;login:

The USENIX Association Newsletter

Volume 9 Number 1 **February 1984**

CONTENTS

News from USENIX 3
USENIX Association Services, Benefits and Membership 3
  USENIX Association Membership Classes — 1984 4
Nominations for Election of Officers and Directors of the USENIX Association 5
Summer 1984 USENIX Conference 6
Call for Papers for the Summer USENIX Conference 7
Call for Papers for the Summer Software Tools Users Group Meeting 9
The First USENIX Computer GO Tournament 10
USENIX Conference Proceedings Available 12
/usr/group Standards Committee Meeting 12
Local UNIX User's Groups 13
Reports on the Presentations at UniForum 14
  Keynote Address 14
  Joint Session 14
  Networking 15
  Distributed UNIX 18
  Compilers and Languages 19
  UNIX Directions 22
  Applications 24
  Implementations I 27
  Implementations II 29
  Communications 34
  Graphics 36
  Real-Time UNIX 39
Information on Vendor Exhibits at UniForum 42
Application for Membership for Calendar Year 1984 71
Release Form for 1984 USENIX Distribution Tapes 73

The deadline for submissions for the April issue of *;login:* is April 16

NOTICE

;login: is the official newsletter of the USENIX Association, and is sent free of charge to all members of the Association.

The USENIX Association is an organization of AT&T licensees, sub-licensees, and other persons formed for the purpose of exchanging information and ideas about UNIX* and UNIX—like operating systems and the C programming language. It is a not-for-profit corporation incorporated under the laws of the State of Delaware. The officers of the Association are:

President	Lou Katz	Directors	Bruce S. Borden
Vice-President	John L. Donnelly		Alan G. Nemeth
Secretary	Lewis Law		Deborah K. Scherrer
Treasurer	Thomas Ferrin		Waldo M. Wedel

The editor of *;login:* is Tom Strong.

Member services are provided through the Association office. Membership information can be obtained from the office:

USENIX Association
P.O. Box 7
El Cerrito, CA 94530
(415) 528-UNIX

Members of the UNIX community are heartily encouraged to contribute articles and suggestions for *;login:*. Your contributions may be sent to the editor electronically at

ucbvax!g:usenix

or through the U.S. mail to the Association office at the address above. The USENIX Association reserves the right to edit submitted material.

;login: is produced on UNIX using *troff* and a variation of the −me macros. We appreciate receiving your contributions in *n/troff* input format, using any macro package. If you contribute hardcopy articles please leave left and right margins of 1" and a top margin of 1½" and a bottom margin of 1¼". Hardcopy output from a line printer or most dot-matrix printers is not reproducible.

This newsletter may contain information covered by one or more licenses, copyrights, and/or non-disclosure agreements. Permission to copy without fee all or part of this newsletter is granted only to Institutional members of the USENIX Association, provided that copies are made only for internal use at the member campus or plant site. No other reproduction of this newsletter in its entirety or in part may be made without written permission of the Association.

---

*UNIX is a trademark of Bell Laboratories.

News from USENIX

Washington DC Meeting

The recent UniForum Conference in Washington DC was the largest UNIX conference held to date on the planet. Over 9000 people paid for admission to UniForum; over one-third of them attended the technical sessions.

The proceedings for UniForum are being prepared by /usr/group. They will include the presented papers from both the USENIX and /usr/group sessions and the information on the vendor exhibits in this issue of *;login:*. Information on ordering your copy is in a following article.

Future Meetings

The next meeting of the USENIX Association will be held June 12−15 in Salt Lake City, Utah. It will be a joint meeting with the Software Tools Users Group. The host will be the Computer Science Department of the University of Utah, with Randy Frank acting as the conference chair. A call for papers and other information on the conference appears in following articles.

Subsequent meetings of the USENIX Association are scheduled for Dallas, Texas, in January, 1985, Portland, Oregon, on June 10−14, 1985, and Atlanta, Georgia, on January 10−13, 1986.

Elections

The election of Association officers and board members will be held this spring. Due to the recent change in the Association bylaws all 1984 members (except student members) will be able to vote, not just the Institutional members. A slate of candidates has been presented by the nominating committee, and in February each of you should have received notification of it and of how to nominate others. A copy of the letter sent to all 1983 and 1984 members regarding the nominations is in a following article.

Statements by each of the candidates will be mailed with the ballots.

Lou Katz, President

USENIX Association Services, Benefits and Membership

A recent change in the bylaws of the Association has allowed some adjustment in the membership categories. The new membership classes and the services and benefits of each class are discussed below. This information and a membership application and tape release form were sent to all 1983 members on February 14. There is also a membership application and tape release form at the back of this issue of *;login:*.

The USENIX Association is a not-for-profit organization of individuals and institutions with an interest in UNIX and UNIX-like systems and the C programming language. It is dedicated to fostering the development and communication of research and technological information and ideas pertaining to UNIX and UNIX-related systems. The Association sponsors semiannual technical meetings, produces and distributes a newsletter called *;login:*, and serves as coordinator of a software exchange via its "Software Distribution Tapes."

The Association began several years ago to meet the needs of UNIX users and system maintainers who met periodically to discuss problems and exchange ideas concerning UNIX. The USENIX Association was incorporated in 1979 to better handle the volume of services needed by its members. It is governed by an elected Board of Directors.

The USENIX Association offers several services to its members:

- Mailing of the newsletter to the members six times a year
- Offering of various UNIX publications and technical information for purchase
- Presentation of technical meetings, single-topic seminars, workshops, and symposia
- A discount on the meeting registration fee for members
- The right to vote on matters affecting the organization, its bylaws, and in the election of its directors and officers
- The right to receive member-contributed software upon completion of appropriate release forms (available only to Institutional and Supporting Members).

USENIX Association Membership Classes — 1984

Any individual or institution with a *bona fide* interest in the activities of the USENIX Association may become a member by filling out an application form and paying the annual fee. Members receive copies of the Association newsletter and other benefits and services enumerated above and below.

Membership is by calendar year. If membership is requested after the first of the year, any previous newsletters (and software distribution tapes for Institutional or Supporting Members) for the year are sent to the new member.

There are four classes of membership, differentiated primarily by the fees paid and the services provided:

STUDENT Membership fee: $15

Open to any full-time student at an accredited educational institution.

INDIVIDUAL Membership fee: $30

Open to any individual or institution. Individual Members receive the newsletter, meeting registration discounts, and may vote. However, they do not receive the Distribution Tape or other services requiring UNIX license verification.

INSTITUTIONAL Membership fee: $100 for educational institutions
$250 for all others

Open to any individual or institution. Institutional Members receive all services available to Individual Members, plus additional services which require UNIX license verification, such as receiving software distribution tapes containing material subject to non-disclosure license restrictions or sending representatives to tutorials or seminars where these restrictions apply. Institutional Members must provide the Association with copies of their UNIX licenses and must sign a USENIX Tape Release Form to take advantage of license-required services.

SUPPORTING Membership fee: $1000

Open to any individual who or institution that wants to support the organization to a greater degree than through the Institutional membership fee. Supporting Members receive all services available to Institutional Members.

Nominations for Election of Officers and Directors of the USENIX Association

The following was mailed to all 1983 members on February 14, and was submitted to net.usenix.

February 2nd, 1984

To: All Voting Members of the USENIX Association
From: Lew Law, Secretary
Subject: 1984 Elections of Officers and Directors of the Association

The biannual election of Officers and Directors of the USENIX Association will be held in the Spring. The Nominating Committee, chaired by Bob Lummis (Albert Einstein College of Medicine) has proposed the following slate:

President	Alan G. Nemeth
Vice-President	Deborah K. Scherrer
Secretary	Lewis A. Law
Treasurer	Waldo M. Wedel
Board Members (4)	Bruce S. Borden
	John Donnelly
	Thomas Ferrin
	Steve C. Johnson
	Lou Katz
	Brian E. Redman
	Michael D. Tilson

The bylaws state that "Nominations for each Office and Directorship may also be made by any five members" (Article 7.1). Members are encouraged to give serious consideration to the representation desired on the Board — as a body it sets the policies and direction of the Association. Should a group wish to nominate a member for election, please send a letter (in writing — no electronic mail!) to arrive before March 1st, 1984 to Lewis Law, 321 Mystic St., Arlington, Mass., 02174. The letter should contain the name of the nominee, a statement that he/she has accepted nomination, and the signatures of at least five members. Nominees should provide a statement, preferably via electronic mail, giving a brief biography and views on the following:

Form of cooperation with /usr/group, if any
Services that the Association should provide to the membership
Structure and format of conferences

These statements will be circulated with the ballots.

Summer 1984 USENIX Conference

The USENIX Association will hold its 1984 Summer Conference in conjunction with Software Tools Users Group, Inc., in Salt Lake City, Utah, June 12–15, 1984. This announcement provides early information about the date of events as well as persons to contact for further information. A pre-registration packet containing detailed Conference information and hotel reservation forms will be mailed in early April.

Meeting Headquarters	Hotel Utah
Exhibit Hall	Salt Palace Center
Conference Host	Department of Computer Science University of Utah
USENIX and Software Tools Tutorials	Tuesday, June 12, 1984
USENIX Technical Sessions	Wednesday through Friday June 13–15, 1984
Software Tools Technical Session	Thursday, June 14, 1984
Paper Deadlines	Abstracts: March 26, 1984 Papers: May 14, 1984
Vendor Exhibition	Tuesday through Thursday June 12–14, 1984
Vendor Exhibit Space Reservations	Mr. Chuck Bierley Industrial Presentations West, Inc. 303-696-6100

The USENIX Association is an organization providing a forum for the exchange of technical information about the UNIX system and associated hardware and software. Such services are provided through sponsoring conferences and workshops, newsletters, publications and software exchange.

Software Tools Users Group, inc. focuses on a set of license-free UNIX-like utilities and system calls, written in ratfor. When running in conjunction with almost any local operating system, the Tools package presents a virtual operating system interface consisting of a virtual machine, utility programs, and a command language, thus achieving inter-system uniformity over a variety of operating systems.

If you did not receive this announcement directly and wish to be on the mailing list for receipt of the pre-registration packet please contact:

USENIX Conference Office
P.O. Box 385
Sunset Beach, CA 90742
213-592-3243

Call for Papers for the Summer USENIX Conference

June 12–15, Hotel Utah, Salt Lake City

Abstracts due March 26

Conference Chair:
Randy Frank
University of Utah

Conference Host:
Computer Science Department
University of Utah

Program Committee:

Jay Lepreau	Univ. of Utah, Co-Chair
Spencer Thomas	Univ. of Utah, Co-Chair
Steve Bellovin	Bell Labs
Kirk Mckusick	U.C. Berkeley
Mike Muuss	Ballistics Research Lab
Dennis Ritchie	Bell Labs

Aspects of this conference which may differ from previous ones include:

- It is sponsored only by USENIX.
- We plan to publish a proceedings for distribution at the opening of the conference.
- We are soliciting formal panels as well as papers.
- We plan to schedule some small semi-formal workshops during the day, and also solicit suggestions for those. They are expected to be somewhat more formal "Birds of a Feather" sessions.
- Papers in the "product announcement" genre will *not* be accepted, and talks of that nature will *not* be tolerated by the session chairs.

Suggested topic areas include, but are not limited to:

- Hardware: evaluations, modifications for the UNIX Operating System, design issues for supporting the UNIX OS.
- Kernel enhancement, performance, real time modifications
- Networks and distributed processing
- UNIX ports and emulations with *new* aspects
- Operating systems architecture: alternate OS's, cacheing methods for hierarchical memories, programming language aspects.
- Programming languages and environments
- Mail, news and *uucp*: transport systems, addressing, routing, user interfaces, USENET problems and solutions.
- Papers involving novel computer graphics or computer aided design topics are especially solicited for a computer graphics session.
- Applications: CAI, System management and monitoring
- 4.2 BSD, System V, 2.9 BSD, GNU: experiences, comparisons, critiques.
- Standards and portability: kernel, commands, library, and languages.
- Security (sic)
- UNIX directions

continued ...

Abstracts of papers should explain what is new and interesting about the work and should be between 300 and 1000 words. Papers previously presented at other conferences will not be accepted. Proposals for panels and workshops should explain the proposed thrust of the discussion, and contain suggestions for at least two members. All submissions must include:

- Title of presentation, panel, or workshop
- Full name of author
- Institution or company
- U.S. mail address
- Network address, if available
- Phone number
- Special audio-visual requirements

Since the program committee plans to produce a proceedings of conference papers for distribution at the conference opening, the following schedule *must* be adhered to:

Abstracts due:	March 26, 1984
Notification of acceptance:	April 16, 1984
Complete papers due:	May 14, 1984

Papers containing no figures may be submitted electronically, *with prior arrangement*. Both Scribe and *troff* will be available, with output to a Xerox 2700 or Versatec. Contact the program committee chairs for more information.

Please be sure that you have the approval of your organization to present and publish your paper before submitting an abstract. A USENIX release form will be distributed with the notice of acceptance, and final papers must be submitted with the completed form in order for us to be able to publish it. The schedule is tight, and does not allow for delays caused by inability to get release approvals. All hard-copy paper submissions must be camera-ready; matrix printer output does not reproduce acceptably, and cannot be accepted.

Abstracts should be submitted to the program chairs at the address below, either by U.S. or electronic mail. Due to the vagaries of the latter we will immediately acknowledge all electronic mail submissions, so try again, or call, if you do not soon receive an ACK.

{harpo,hplabs}!utah-cs!usenix or usenix@utah-cs.ARPA

Jay Lepreau or Spencer Thomas
Computer Science Dept.
University of Utah
Salt Lake City, UT 84112
(801) 582-7214, 581-3095, 581-4285

Call for Papers for the Summer Software Tools Users Group Meeting

June 12—15, Hotel Utah, Salt Lake City

Abstracts due March 23

Software Tools Users Group, Inc. will be holding a meeting in conjunction with the USENIX Association's meeting. Individuals interested in making a presentation at the Software Tools Technical Session are invited to submit abstracts by March 23, 1984, to the address below.

Suggested Presentation Topics:

- Applications using the VOS
- Programming Environments
- Software Development Tools
- New and Enhanced Utilities
  - Graphics Tools
  - Networking Tools
  - Text Processing Tools
- Software Standardization and Portability
- Future Directions

Abstracts

Abstracts *must* be submitted by March 23, 1984. They should explain what the work is about and should not exceed 1000 words. Abstracts must contain the following information:

Title
Name of author
Institution or company name
U.S. Mail address
Phone number (and network address, if available)
Audio-visual requirements

Abstracts should be mailed to:

Barbara Chase
S.T.U.G. Conference Coordinator
9619 Belford Ave, #3
Los Angeles, CA 90045

chase@lbl-csam
(213) 647-0272, 641-5434

Proceedings

A conference proceedings consisting of papers on the subject of the presentations will be available at the conference. Therefore, the presentation papers *must* be submitted to the above address by May 4, 1984. Paper submissions are not required by Software Tools; however, they are highly recommended. These paper submissions must be camera-ready; matrix printer output will not reproduce acceptably.

The First USENIX Computer GO Tournament

To be held during the Summer 1984 USENIX conference in Salt Lake City, Utah, June 12–15.

```
   ABCDEFGHJKLMNOPQRST
19 +++++++++++++++++++ 19
18 +++++++++++++++++++ 18
17 +++○●++++++++++++++ 17
16 +++○+++○+●+++++●+++ 16
15 +++++++++++++++++++ 15
14 ++○○+++○+●+++++++++ 14
13 ++●++++++++++++++++ 13
12 +++++++++++++++++++ 12
11 +++++++++++++++++++ 11
10 +++++++++++++++++++ 10
 9 +++++++++++++++++++ 9
 8 +++++++++++++○○○○●+ 8
 7 ++○●+++++++++○●●●●● 7
 6 ++●○○+++++++++○○○●+ 6
 5 ++○++++++++++++○●●+ 5
 4 +++○+++++++++++○●++ 4
 3 ++●●+●++++++++●●○●+ 3
 2 +++++++++++++++++++ 2
 1 +++++++++++++++++++ 1
   ABCDEFGHJKLMNOPQRST
```

Most Probable Rules

1. A full size board will be used.

The board will be 19 x 19. The columns will be labeled ‘A’ through ‘T’ (excluding ‘I’) left to right. The rows will be labeled ‘19’ through ‘1’, top to bottom.

2. A “referee” program enforces syntax.

The tournament will use a “referee” program to execute each competing pair of programs; thus the programs must understand a standard set of commands and generate output of a standard form.

All input commands to the program will be in the form of lines of text appearing on the standard input and terminated by a newline.

a) The placement of a stone will be expressed as letter-number (e.g. “G7”). Note that the letter “I” is not included.

b) A pass will be expressed as “pass”.

c) The command “byo-romi” means the time limit has been exceeded and all further moves must be generated within the shorter minimum time limit.

All output from the program will be in the form of lines of characters sent to the “standard output” (terminated by a newline) and had better be unbuffered.

a) The placement of a stone will be expressed as letter-number (e.g. “G12”). Note that the letter “I” is not included.

b) A pass will be expressed as “pass”.

c) Any other output lines will be considered garbage and ignored.

3. The "referee" program has other duties.

4. The "referee" program has other duties.

Aside from timing the competing programs and passing the moves back and forth, the referee program will maintain a display of the board, the move history, etc.

5. "Bad" moves are a forfeit.

Any syntactically correct but semantically illegal move (i.e. spot already occupied, *ko* violation, or suicide) will be considered a forfeit.

6. There will be a time limit.

Each program will be limited to a total of 60 minutes of accumulated "user" time. If a program goes over the time limit it will only be allowed 10 seconds of "user" time for each move (*byo-romi*). If a program goes over the time limit and uses more than 10 seconds of "user" time for a move it will immediately forfeit the game.

7. Play will continue until both programs pass in sequence.

Although this may be a trouble spot, it looks like the best approach available.

8. The Judges decisions will be final.

A human judge will evaluate the game results and may fill in missed *dame* or may judge a game incomplete if too much is unresolved.

9. Japanese rules will be used, (Nihon Kiin).

10. Komi will be 5.5 points.

The second player gets a 5.5 point bonus.

11. Computing resources may be limited.

The general form of the tournament will depend on the number of participants, the availability of computing power, etc. If only a few programs are entered each program will play every other program once or twice. If many are entered some form of elimination system will be used.

12. These rules are not set in concrete.

Except this one, of course.

Comments, suggestions, contributions, etc. should be sent via *uucp* to harpo!psl or via U.S. Mail to

Peter Langston
Lucasfilm Ltd.
P.O. Box 2009
San Rafael, CA 94912

USENIX Conference Proceedings Available

Washington DC UniForum Conference — Winter 1984

The proceedings of the January UniForum Conference are being produced by /usr/group and will be available in June or July. The price is $30 per copy, plus $15 per copy for overseas postage. They may be ordered from:

/usr/group
4655 Old Ironsides Drive
Santa Clara, CA 95050

Toronto Conference — Summer 1983

Copies of the proceedings of the Toronto Conference are available for $30 per copy, plus $15 per copy for overseas postage. Send your check or money order to:

USENIX Association
P.O. Box 7
El Cerrito, CA 94530

San Diego UNICOM Conference — Winter 1983

Copies of the proceedings of the San Diego UNICOM Conference are available for $25 per copy, plus $10 per copy for overseas postage. Send your check or money order to:

Software Tools Users Group
1259 El Camino Real, #242
Menlo Park, CA 94025

/usr/group Standards Committee Meeting

The /usr/group Standards Committee met on Tuesday, January 17 at the UniForum Conference to review the latest version of the Standards document, which had been edited to incorporate changes agreed upon at the previous committee meeting held at Comdex on Tuesday, November 29 in Las Vegas. No major changes were made at the last meeting. After much discussion, a motion was made to submit the Standards document to the general /usr/group membership for a vote on the adoption of the Standard as it now exists. This was passed by a vote of 24 to 3. The Standards document had previously been submitted to the /usr/group membership for comments.

After a final round of editing and phototypesetting, the Standards document, along with the Reviewer's Guide, will be sent to the /usr/group membership for a vote. This is expected to happen towards the end of February. A two-thirds vote is required for adoption of the Proposed Standard. The documentation will include a comparison with System III and System V, as well as a detailed description of the file/record locking primitive so that members can vote with access to the proper background information.

In further discussions held on January 17, extensions in the areas of terminal handling, networking, real time, interprocess communication and object file formats were considered. It was felt that something had to be done about the *ioctl* system call so that the terminal control capabilities could be specified. After some debate, it was agreed that the System V version of *ioctl* would be the best starting point. A proposal will be generated for consideration at the next /usr/group Standards Committee meeting, to be held in April in conjunction with the Winter Comdex meeting in Los Angeles.

The /usr/group Standards Steering Committee feels that the *proposed* Standard has now received adequate review and is ready for a public vote. The adoption of this Standard will benefit both suppliers and users of UNIX-based and UNIX-like systems. Copies of the Standards document are available for $50 from

/usr/group
4655 Old Ironsides Drive
Santa Clara, CA 95050
408-986-8840

If you would like to participate in the /usr/group Standards effort, or would like to make a proposal, or even review an existing proposal, please write to the Chair of the Standards Committee:

Heinz Lycklama
Interactive Systems Corp.
1212 Seventh Street
Santa Monica, CA 90401

Local UNIX User's Groups

The USENIX Association will publish information on local user groups in *;login:*. Information on local groups giving the name, address (phone number and/or net address is desired), time and location of meetings, special events, etc. is welcome. Please call the USENIX office if you wish to have your group added to the list. Our current list of local groups follows.

The Front Range group meets about every two months at different UNIX sites for informal discussions.

Front Range Users Group
N.B.I., Inc.
P.O. Box 9001
Boulder, CO 80301
Attn. Wally Wedel
(303) 938-2923

Dallas / Fort Worth UNIX User's Group
Advanced Computer Seminars
2915 L.B.J. Freeway, Suite 161
Dallas, TX 75234
Attn. Irv Wardlow
(214) 484-UNIX

There is an informal group that meets in the Washington, D.C., area every two months or so. The current contact for that group is:

Neil Groundwater
Analytic Disciplines, Inc.
Suite 300
8320 Old Courthouse Road
Vienna, VA 22180
(703) 893-6140
npg@lbl-csam

Unigroup is a non-profit organization in the New York City area for users and vendors of products and services for UNIX systems. It plans to publish a quarterly newsletter, a local *uucp* directory, and a directory of local companies offering UNIX system products and services.

Unigroup of New York
G.P.O. Box 1931
New York, NY 10116

Reports on the Presentations at UniForum

These reports were prepared by Karen Summers-Horton with the assistance of Jerry Deroo and Berry Kercheval. Reporter and editoral comments are enclosed in square brackets.

Keynote Address

Reporter: *Karen Summers-Horton*
2843 Valcour Court
Reynoldsburg, OH 43068

Keynote Address

Jack Scanlon
AT&T

Mr. Scanlon spoke on the history of UNIX and the present status of the UNIX revolution. The early conferences in the mid-70's led to the acceptance of UNIX by the university communities. The first commercialization came in the mid-70's by AT&T, and in 1975, System III standardized UNIX.

Today, 70,000 computers run UNIX, with over 300 applications packages available. Mr. Scanlon went on to outline some pressing market needs that exist today. UNIX is moving to versatility, with portability to several different hardware types.

In the short term, there is a need to increase the switch to System V by independent vendors, both high and low end. Mr. Scanlon restated AT&T's commitment to support of System V.

Joint Session

Reporter: *Karen Summers-Horton*

The presentation of the /usr/group corporate sponsor awards went to AT&T, Plexus Computers, Gould, Inc., Zilog, Pyramid Technology, Onyx Systems, Human Computing Resources, Wollongong Group, and Santa Cruz Operation. There were also seven additional new corporate sponsors.

It was announced that the /usr/group Standards Committee had made its final acceptance of the proposed standard for UNIX and UNIX-like systems and that the proposed standard was now being submitted to the general /usr/group membership for a final acceptance vote.

Lou Katz of the USENIX Association announced that the recently voted on bylaw revisions has been accepted. The revised bylaws eliminate ambiguities and restructure the membership classification so all but student members have the right to vote. The next USENIX conference will be in Salt Lake City, Utah, June 12→15, 1984. The sites for future conferences will be: Dallas, Texas, in January, 1985; Portland, Oregon, on June 10–14, 1985; and Atlanta, Georgia, on January 10–13, 1986.

The president of the European UNIX Users Group (EUUG), Emrys Jones spoke on EUNET. One of the services offered by EUUG is a twice annual conference (the next to be April 16–18, 1984 in Nijmegen, Holland). EUNET currently consists of 121 sites. The EUUG structure is such that all of the groups elect a board, of which one member serves on a national board. This board, in turn, elects five members to an Executive Board. The national board has control and jurisdiction over EUUG via the main board member, but EUUG has virtually no control over the national group.

Next, Tom Crowley of AT&T Technologies spoke on AT&T's new products. He restated AT&T's commitment to the support of UNIX System V. Several new products include:

- UNIX, System V, Release 2 (which will be upward compatible).
- UNIX Documenter's Workbench Software (complete text processing capabilities; it is included in Release 2 upgrade for present licensees).
- BASIC Interpreter (compatible with IBM PC Basic).
- MC 68000 Software Generation System (current compiler technology).

Mr. Crowley also spoke of AT&T's new business policies. This includes Level One support for source licensees. This is three months support at no charge with UNIX System V, Release 2, and is available for many processors. Another announced new policy is royalty-free use of run-time libraries for applications developers. There is also more flexible royalty schedules for System V resellers.

Networking

January 18, 1984, 1:30 p.m.
Chair: Thomas Ferrin, University of California, San Francisco

Reporter: *Jerry Deroo*
4075 Pheasant Run
Mississauga, Ontario
Canada L5L 2C2

Driver-Based Protocol Implementations

Greg Chesson
Silicon Graphics
630 Clyde Court
Mountain View, CA 94043

Greg's contention is that UNIX has, until now, stood in the way of network implementations. Networks are now supported by hardware and evolving software. So why not simulate "future" hardware by hiding a layer of software in a classic device driver? This would minimize the complexity of the operating system interface.

The goals of the project were stated as:

- implement a LAN which requires a device driver to implement the "classic" network
- have a simple path for distributed files
- allow bulk transfer
- implement DMA.

They have implemented the socket, or character device side. They have not quite got the block device side running.

They implemented this with about 1200 lines of code in the driver, 1000 lines to implement data transport, and a further 600 lines of code to "glue it together." The system:

- has good character I/O rates
- has no problem with typed lines
- can support block I/O.

They have implemented this on all versions of UNIX from V7 up. To access a virtual terminal, all you do is point *gtty* at a special file.

They have found that not all network protocols work well with this approach. Very complex ones would tend to over-load the system, as much of the protocol code is executed at interrupt time.

They found that UNIX helped, more than hindered, their development efforts.

They would like to suggest that while networks have come a long way, that "technical responsibility is NOT the market driver." They suggest that the "gotcha principle" is still there, whereby once you have chosen a technology, you are pretty well locked into it.

They conclude that

- IPC is not required for a network
- files, terminals and other network objects can be accessed through a device driver
- hardware developments will permit further software simplification as functions can be moved from driver to board level
- standards are still incomplete

The Excelan TCP/IP Protocol Package

Bruce Borden
Silicon Graphics
630 Clyde Court
Mountain View, CA 94043

Bill Northlich
Northlich Computer System & Software
21 Newell Court
Walnut Creek, CA 94595

This talk described the implementation of the DOD standard protocol IP/TCP on an intelligent Multibus Ethernet front-end controller. This move, which requires about 35 Kb on the board, has resulted in a significant off-loading of the host. It simplifies the host/network interface, and improves the portability of the protocol.

They chose TCP/IP for the following reasons:

- it is fully specified
- it is an accepted standard
- it is widely implemented
- it has Berkeley support

They have observed a transfer rate of 12 Kb per second [?] in process to process transfers, and 12 Kb/s in file transfer. The implementation results in the fact that a file transfer over the network takes less system overhead than a local file copy.

There were some complexities introduced into the effort due to hardware peculiarities. They first had to simulate the "inside" of UNIX, to allow them to put the 4.1cBSD version into the board.

Current work is covering these areas:

- profiling the board for tuning
- allowing more than one board on a host
- internetworking
- modifications to go to 4.2BSD
- statistics gathering
- place telnet/rlogin into the board

They are currently able to interface with V7, System III and System V.

Worknet: A Xenix-Based Merged Filesystem Network

Alan Greenspan
Altos Computer Systems
2641 Orchard Park Way
San Jose, CA 95134

This talk covered the high-level design and implementation of their network. It uses the RS-422 and Ethernet transport levels, and can support up to 30 terminals with a maximum distance of 500 feet [not sure if the distance is total or node-to-node].

The implementation places one more tree level into the UNIX file structure, a "super-root" directory, from which machines on the network are accessed as "directory" names. This results in a network model consistent with the model of a single UNIX machine's filesystem.

With the implementation, it is possible to do the following:

- access a remote raw device
- move the current working directory to any other machine
- remote execution of an command. This results in non-network qualified pathnames to be relative to the executing machine.
- with symbolic links, mail, *lpd* and other services can be centralized.
- pipes can span across the network.

CSNET Grows Up

Michael T. O'Brien
The Rand Corporation
1700 Main Street
Santa Monica, CA 90406

Daniel B. Long
Bolt, Beranek and Newman, Inc.
10 Moulton St.
Cambridge, MA 02238

This talk was a "state of the product" discussion. It was the contention of the authors that CSNET was a means by which cost-effective access to Arpanet could be provided. The development group is maturing, and expect to move to a production environment in three years. Currently, CSNET has automated 80% of the site management task.

During development of the system, and in trying to widen its user base, the authors have run into several user problems:

- confusing, and varied, path conventions
- how to find a specific person
- message headers that are not too understandable
- user expectations of the system.

They want something which offers

- the cost of the post office
- the ease of use of the telephone
- and the speed of the telegraph.

It is difficult to meet all of these criteria.

Distributed UNIX

January 18, 1984, 3:30 p.m.
Chair: Alan Nemeth, Prime Computer, Inc.

Reporter: *Karen Summers-Horton*

Software Administration Over Computer Networks — The exptools Experience

Joseph L. Steffan
AT&T Bell Laboratories
Naperville, IL 60566

The experimental tools package (exptools) at AT&T Bell Laboratories in Naperville, Illinois is administered and distributed via computer networks. The exptools package includes over sixty tools, maintained by about twenty people on more than one hundred machines of different types connected by two different networks. There is one coordinator, who controls the division of the workload and the installation of the tools on the various machines.

There are many obvious benefits to this system. It reduces the computer centers work, and enables the tools to be updated fast (a matter of days instead of weeks). Also, one person can administer many machines, because it does not significantly increase his workload.

There have been some operational problems, including the network being down to one or more machines, network jobs disappearing, files being truncated on delivery, and the target file systems being out of space.

There is also the question of security. The system is probably secure enough for the corporate environment, but there is concern about whether or not it is secure enough for the college environment.

In conclusion, it is believed that this system points the way to administration of hundreds of desk-top computers connected with a network, and removes a stumbling block to their acceptance.

A Distributed File System for UNIX

Matthew S. Hecht, John R. Levine, Justin C. Walker
INTERACTIVE Systems Corporation
8689 Grovemont Circle
Gaithersburg, MD 20877

The standard topic for this session seems to have been remote filesystems for UNIX. This was one of the more scholastic of the papers presented. They described a way to share filesystems among homogeneous UNIX systems.

Similar work cited included S-UNIX/F-UNIX at Bell Labs, and a Plexus design by Groff. The LOCUS project at UCLA is more ambitious, because it deals with replicated files and transactions. The COCANET project at Berkeley is also more ambitious since it handles remote processes.

The solution is based on a "remote mount" primitive, that can mount a remote disk onto a local file tree. The work is split across the two machines at the inode level — the client does a remote procedure call of the *nami* function, which is carried out on the server.

Problems that are not solved include dealing with "..", loops in *rmount* structures, and security. No provision is made for sessions or restart detection.

No hard performance data is available. It was expected that access to remote files would be about one half as fast as local files.

Multiprocessor Debugging on a Shared Memory System

Chet Britten, Paul Chen
Metheus Corporation
5289 NE Elam Young Parkway
Hillsboro, OR 97123

This talk described the debugger used to help implement the Metheus Lambda 750 VLSI design workstation. The workstation uses two processors, and a special purpose debugger was needed. The workstation is based on 4.1BSD with Ethernet, virtual memory, color graphics, multiple windows, multiple processors, and an OEM Omega display controller.

The system was ready for beta test in 12 months. The software was done by three software engineers. The debugger more than paid for itself in development time it saved.

Some of the problems encountered were shared vectors, processor identification, different traps on different processors, changing the trace vector, the ID register, unused bits, and stack range.

Some improvements made include an option to continue tracing; indirect addressing capability; counting the number of breaks or traces; knowing about structures; having only the correct processor stop; knowing about the memory map for the user program, which is harder with virtual memory; breaking on a virtual address, which has to know a lot about UNIX; providing a logic analyzer type trigger, to trace instructions on a page fault; and nested triggers.

Having UNIX has proved to be an advantage. It provides job control, shell escapes, and the *nlist* utility.

Panel Discussion: Multiprocessing Issues

The panel consisted of:

Joseph L. Steffan, AT&T Bell Laboratories
Chet Britten, Metheus Corporation
Paul Chen, Metheus Corporation
Monty Picard, Plexus Corporation
Greg Chesson, Silicon Graphics

Issues discussed included lightly coupled versus tightly coupled systems, and the advantages to different approaches tried.

Compilers and Languages

January 19, 1984, 8:30 a.m.
Chair: Louis Salkind, New York University

Reporter: *Jerry Deroo*

The Evolution of the Portable C Compiler

S. C. Johnson, L. Rosler
AT&T Bell Laboratories
Murray Hill, NJ

The original *pcc* was built in the mid 70's. It has been moved to over 30 different machines. In so doing, it has become an effective C standard. The compiler itself has become reliable as much of the code has been re-used in most of the ports.

However, the original

- was slow
- assumed a limited target machine architecture; for example, no provisions for three address instructions, or more complicated address modes than the old reliable, PDP-11
- the porting process was complex. It involved hand-writing many non-trivial C functions to do the code generation. The process was never designed, it just evolved.

Over the last few years, effort has been made to improve the package. The porting process has been better designed. Though not yet a cookbook, it is table driven. The porting process is based on recent theory. The machine description is in a table. The code generation portion is now mostly automated.

They found that the theory does not handle some hardware features very well. Examples cited were:

- register pairs
- condition codes
- multiple register types
- ''strange'' op-codes

They have made some policy decisions regarding how to implement various features. If there are only two or three choices for how do do something, the choice is controlled by #ifdef's in the compiler. If there is only one way do do something, that way is provided. A user can, of course, roll their own way.

Now that the code generation part of the process of generating a new compiler is simplified [!], there are still some hard-to-implement areas. Examples of these are:

- stack frame layout
- switch code generation, struct copy
- the interface to the assembler
- width and alignment of data-types
- register variables, and register allocation

Developing and testing a new compiler is now a matter of:

- picking a similar target machine, and modifying it. This is possible now that a few machines have been done. This usually takes less than one week.
- the development of the assembler and the loader now dominate the time allocation.

They have, so far, generated compilers for the 68000, and for the VAX. One nice thing that is now possible to do as it is easier to change things, is to build a version of the compiler that will generate statistics about the program it is compiling.

They are looking to use this new version for leverage in the continued struggle to improve the compiler. It has provided a foundation onto which they can add more tools for compiler development. As the compiler is less monolithic, they can clean up ancient code.

Future development will include changes to the language's already renowned error recovery capabilities, the implementation of classes, and the construction of the second generation version of *lint*. Furthermore, they see evolution in the program development environment, including things like a super-*make*, a syntax-directed editor, and support for incremental compiling and debugging.

How to Feel Better About Knowing It is Written in C

Alan R. Feuer
Catalytix Corporation
Cambridge, Massachusetts

The speaker contends that critical programs that are written in C are perceived as dangerous. While UNIX is great for a production software development environment, other environments are better at "some things."

Contention: it should be possible to develop software in C with the safety of Pascal without sacrificing the flexibility of C. The speaker then gave several code examples of problems which the compiler believes, but execute incorrectly. In fact, some of the examples got by *lint*. *lint* can help, but is not enough.

The speaker has implemented a compiler which can place in line run-time code into the program. This would be used mainly during the development of a program, and be turned off once it was ready for production.

The speaker feels that C embodies some old assumptions which now need re-examination. [He did not have time to re-iterate those.]

An Implementation of the B Programming Language

Lambert Meertens, Steven Pemberton
Centrum voor Wiskunde en Informatica
Kruislaan 413
1098 SJ Amsterdam

"The name of the language is called B." Having said that, the speaker stated the design objectives of the language. They were:

- simplicity
- suitable for conversational use
- a tool for structured programming

They have designed the language by iteration, and the language has evolved to one for personal computing. The language is easy to learn, as it has few constructs. It is easy to use, as it is powerful. These "pluses" for the user have made the language not easy to implement for the designer. [Then again, is that not what it is all about.]

The language is command driven. New commands can be added, which allows the development of subroutines and functions. The language is strongly typed, yet variables need not be declared. Their type is inferred from the context in which the variable is used. String operators are built into the language.

The first version was implemented, in C, with about two person months of effort. It was not efficient, but was usable. Linear lists were used for storage. The next version entailed about four person years of work. Attention was paid to efficiency. The implementation was more modularized. B trees were used for the storage management.

The language is interpreted. It embodies an editor which does the syntax checking as the code is being entered. The editor supports command completion.

The product runs under UNIX. It has run on the VAX, an 11/45, a PERQ and a 68000-based machine. The product is available for a nominal cost.

Future work will address

- a fully integrated environment

- more efficiency
- static semantic checks
- graphics support

Object-oriented Programming in C Language

Brad J. Cox
Productivity Products Int.
37 High Rock Road
Sandy Hook, CT 06482

The speaker described the reasons for chosing to implement some extensions to the C language from Smalltalk-80.

With these extensions, it is now possible for one person to develop, in the same time, what using traditional methods would require several people. This is possible because:

- the extensions enhance the reusability of the code that is written
- the extensions make handling abstractions easier, so the "user" is more efficient
- the code is more changeable.

The speaker described the implementation of a tool, a "Screen Application Generator," which was built to generate a model application, an expense account information collection program. The screen application generator, SAG, is given a description of the application and generates C code. This code is combined with a large library of support functions, which are re-usable higher-level functions — an applications library.

This tool allows an application to be described as a hierarchical group of processes, in which lower related processes inherit the context of their "parent."

The speaker stated that this tool results in the reduction of the amount of code which must be generated.

UNIX Directions

January 19, 1984, 10:30 a.m.
Chair: B. E. Redman, Central Services Organization

Reporter: *Karen Summers-Horton*

Behind Every Binary License is the UNIX Heritage

B. E. Redman
Central Services Organization

P. E. Parseghian
AT&T Bell Laboratories

The philosophy of UNIX was discussed: how did it come about, and why did it grow to the state it's at now? The development of UNIX took place over a period of time, and its purpose was to create an environment that would be convenient to program development. Today, there is a new concept of UNIX: the academic UNIX of the 70's has faded away and commercial UNIX is here ("The Cheese").

Money has become very important, and competition is marking the development of UNIX today. The authors are troubled by the "unbundling" of UNIX — each part becoming an option. As more different types of UNIX appear, their value diminishes. Portability has led to incompatibility.

In order to preserve the essence of UNIX, there needs to be two separate UNIX's — the academic UNIX and the commercial UNIX. The authors believe the academic UNIX can be preserved, and the commercial UNIX is not appropriate for their needs. There needs to be a reconsideration of the development of UNIX, and the development of another UNIX, modeled after academic UNIX.

The high visibility and enormous growth of UNIX has taken place largely due to the fact that UNIX is less rigorous, less uniform and less antiseptic. Stability, clarity and uniformity have emerged, and in the authors opinion "paradise is boring."

How Did UNIX Get Here? A Sketchy History of the UNIX Development

Andrew Tannenbaum
MASSCOMP

We can view UNIX as the "Cabbage Patch" operating system. It is the latest craze, but how did it get here? UNIX was always a telecommunications operating tool — AT&T was not allowed to compete so UNIX grew up in unusual circumstances. It was lying in a dormant state, waiting for the market to explode.

The future of UNIX was controlled by USG (UNIX Support Group) inside AT&T, and they controlled the future directions of UNIX through the 1970's. UNIX was originally viewed as an interface to the big machines that would do the heavy number-crunching. Later, the University of California, Berkeley and ARPA would have different ideas. It is important to note that USG was a SUPPORT group, and there was no UNIX development group.

When the VAX 11/780 came about, Berkeley emerged as the system of choice. U.C.B. UNIX was considered more programmer friendly. USG's attitude was that Berkeley UNIX was "not really faster, it just seems faster."

In conclusion, the author believes that Bell never really made efficient use of their resources in relation to the development of UNIX.

A Proposed Syntax Standard for UNIX System Commands

Kathleen Hemenway, Helene Armitage
AT&T Bell Laboratories
Murray Hill, NJ

Past attempts to standardize the syntax of UNIX system commands have not been successful. The authors believe that times have changed, and there are several reasons for acceptance of this standard: there is greater recognition of the importance of user involvement; applications and tools are being developed in more different places; standardization has become a part of the UNIX system world; and there is more concern about the evolution of utilities.

There was a practical concern to be consistent between the proposed syntax and the existing syntax. There were also several technical concerns including compatibility with alternative shells, generality and robustness, and the idea that the syntax should be as simple as possible.

The method used to develop the standard involved three phases. Phase one was a conceptual and statistical analysis of the syntactic structure of the command set. Phase two involved an evaluation of technical and human factors aspects of alternative syntax rules. In phase three the strengths and weaknesses of alternative rules were weighed against the impact that selecting each rule would have on the command set.

A total of thirteen syntax rules were developed:

Rule 1: Command names must be between two and nine characters.

Rule 2: Command names must include lower case letters and digits only.

Rule 3: Option names must be a single character in length.

Rule 4: All options must be delimited by "-".

Rule 5: Options with no arguments may be grouped behind one delimiter.

Rule 6: The first option-argument following an option must be preceded by white space.

Rule 7: Option arguments cannot be optional.

Rule 8: Groups of option-arguments following an option must be separated by commas or separated by white space and quoted.

Rule 9: All options precede operands on the command line.

Rule 10: "--" may be used to delimit the end of the options.

Rule 11: The order of options relative to one another should not matter.

Rule 12: The order of operands may matter and position- related interpretations should be determined on a command-specific basis.

Rule 13: "-" preceded and followed by white space should be used only to mean standard input.

It is hoped that these standards will preserve the time-honored features of the UNIX system and actively guide the evolution of the utilities. It will also, inevitably lead to more standards and guidelines.

Panel Discussion: UNIX Directions

The panel consisted of:

Brian Redman, CSO
Rob Pike, AT&T Bell Laboratories
Mike O'Dell, Group L
Armando Stettner, Digital Equipment Corporation
Andy Tannenbaum, MASSCOMP
Ed Gould, mt Xinu
Mike Karels, University of California, Berkeley

Applications

January 19, 1984, 1:30 p.m.
Chair: Reidar Bornholdt, Columbia University

Reporter: *Jerry Deroo*

MLE (Multi-Language Editor)

Scott Bradner
Harvard University

MLE is an editor for natural languages. It was designed and implemented as a support tool for the humanities.

Some of the difficulties which have to be addressed when designing and implementing such a product are:

- difficulty of representing a language's character set on terminals
- designing an internal representation
- some languages have more than 100 characters; northern European languages have upwards of 200 characters

- accent representation is a problem; even more difficult is placing two accents on the same character

To facilitate the user, several features have been implemented. The keyboard is completely mapable, so that an user can configure the keyboard as they desire. The direction of cursor motion is modifiable, to suit the situation.

Each character is represented in a long integer, which contains all the information required to draw the character. Manuscripts which are imported into the system's database are now translated into the system's representation each time they are accessed. They switched to this method after finding that one-time processing made it difficult to handle the text. If they ran into an inconsistency in the translation, it might go undetected for some time.

The system supports typesetting of the languages. *ditroff* has a separate hyphenation algorithm for each language. *sort* knows about the sequence of characters in each language.

They are working on spelling checkers for each language. They feel that the package is not yet ready for distribution. They expect to have the full implementation of the editor done in June. They are doing some work on doing the fonts on a bit-mapped graphics terminal.

MPS: A UNIX-Based Microcomputer Message Switching System

T. Scott Pyne, Joseph S. D. Yao
Hadron, Inc.
6th Floor
1945 Gallows Road
Vienna, VA 22180

The speakers described their implementation of a message switching system used for communication between a base unit and mobile vehicles. Their mandate was to replace an existing hardware configuration, while retaining the existing mobile units. Their system had to "drop-in."

Due to performance considerations, and timing problems, they did not want to use pipes for inter-process communication. They did not, however, have access to a system which supported ports. They wanted shared memory, which was also not supported. Given that all they had was a binary Xenix, what to do?

They implemented shared memory by patching the kernel to put in their modifications. They implemented ports without having to make any kernel modifications. They used the Rand named-pipe approach, but had to modify it as they could not make any kernel modifications for this. The constraint is that the reader process must keep up to the writer, as there is no write blocking. They are using fixed-size messages.

Their modifications will work on systems from Version 6 up. Currently, their code is being handled by lawyers to determine distribution requirements.

Taming the Beast (An RSX Emulator for UNIX)

Daniel R. Strick
University of Pittsburgh

This package allows a binary built on an RSX system to be executed on a UNIX system.

The speaker was faced with a dilemma: he had a commercial RSX product, and obtained a UNIX system. What should be done with the commercial product?

He needed to be able to construct RSX binaries on UNIX, so he needed an interface which supported the RSX system interface.

The speaker contends that "an emulator for system X should make data translations implemented by the system X binary available on UNIX." This requires that:

- UNIX files can be accessed as UNIX files and as RSX files
- RSX files can be accessed as both UNIX and RSX files
- the emulator process must be runnable in a pipeline

Some of the obstacles which had to be overcome:

- file name mapping: RSX is a flat file system
- file attributes: RSX has a lot of them
- file types: RSX has these; this requires re-formatting from one type of file storage to another on the fly, depending on how the file is to be accessed.
- there is some similarity between RSX and UNIX in the file system representation, the difference being that RSX speaks in terms of a file number, while UNIX speaks in terms of a file name.
- RSX can not handle more than one file name (pathnames), which results in the path */usr/dan/junk* being treated as "file junk on device /usr/dan."

A UNIX Based Color Graphics Workstation

Rex McDowell
Metheus Corporation
Hillsboro, OR

The speaker described the design and implementation of a graphics workstation. The machine is built around three 68000 CPUs and a bit slice display processor. The machine runs 4.1BSD.

Each window on the screen is treated as a separate tty by UNIX. *termcap* has been modified to allow the colors of the window to be accessed. This required changes to login, and some low level tty driver changes. The activation of a new window asserts carrier on for the tty.

Window overlay is done by twiddling the color maps. Pop-up menus are stored in off-screen display memory, and copied in when they are needed.

They have implemented a core graphics library, which is a subset of the CORE standard. The system supports the notion of a "graphics entity," which is a display item. The display processor can do scaling and rotation on these entities.

User-Level Window Managers for UNIX

Or: Text Windows Without Kernel Modifications

Robert J. K. Jacob
Naval Research Laboratory
Washington, DC 20375

The speaker has implemented a window management facility that does not require any modifications to the UNIX kernel. It uses facilities that have been supplied by the kernel, from as far back as Version 6, to implement multiple windows on a screen.

The user can move from one window to another at will. Each window has a shell running in it, with a *termcap* entry which describes its size.

The package is implemented as a series of processes connected with pipes. One process listens to the keyboard and directs the input to the appropriate shell controller process, while also sending the input to the display management process, which does the echoing. Each shell control process feeds its shell, and returns characters from the shell to the display manager. The entire package is based on the assumption that everything could be represented as a character stream, and it did not matter where the stream came from or went to. (Things like *vi* fall down here.) Each window emulates a glass terminal.

Versions of the package that run on more recent versions of UNIX can take advantage of some of the new features.

The software has been posted to the net.

Implementations I

January 19, 1984, 3:30 p.m.
Chair: Michael O'Dell, Group L

Reporter: *Karen Summers-Horton*

The UNIX Paging System

Keith A. Kelleman
AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974

Work is currently underway at AT&T Bell Laboratories to develop a demand paged kernel for UNIX System V. It is hoped that it will utilize the best features of both 32V and BSD.

Some of the requirements for the System V paging is that it will require no program changes — it will not change system calls. A priority is to not hurt those who don't need paging — there should be no performance loss. Also, as part of the development, there will be a definition of memory management architecture — the architecture must be general enough to support many different memory management units, paging and swapping kernels. A main component of the architecture is the "region." A region is a data structure within a system that represents a potential virtual address space.

The reason for not merely supporting Berkeley paging is because AT&T feels it is too complex and too specific to the VAX (not portable enough). They also do not like the *vfork* call, and claim there is no well-defined architecture.

The status of the project is that there is currently a prototype implemented on the 3B20 computer. The performance is good relative to System V, Release 2. They are currently porting it to a VAX.

Providing a Job Control Facility in UNIX System V

W. P. Weber, Jr., L. S. Weisbrot
AT&T Bell Laboratories
Murray Hill, NJ 07974

The reasons for wanting a job control facility are numerous. They include the limitation of a single terminal, and the sequential nature of the single terminal. BSD solved some of the problems of job control. The authors want the ability to layer the terminal and have a controller that switched them between the layers.

The implementation took place in two parts: the kernel and user pieces. The kernel piece consists of a driver, *sxt*, that provides mapping between multiple virtual terminals and single real terminal. The user piece consists of a user-level utility, *shl*, that provides control through layer control and manipulation.

This model supports the definition of job control, providing multiple environments. The authors see it as "the poor persons Blit." Some distinct advantages are that no changes are required to existing programs to run under the facility, the impact on the kernel is small, and no changes are required to the shell.

Reloadable UNIX Device Drivers

Thomas A. Alborough
Creare R&D Inc.
Box 71
Hanover, NH 03755

This talk was different from the announced talk ("Loadable Virtual Disk Device Driver and Server") because there was insufficient time for a good presentation of the original topic. According to the abstract, the original talk dealt with a local device driver that provides transparent access to a remote device.

They needed to develop a device driver in the UNIX environment. The objective was to reduce development time and provide a better product. The technique used was to overwrite the old driver code and data with new.

Elements of the technique: the driver code and data are of fixed size. The driver code and data are locatable (via *nlist*). The driver vector entry points are vectored, making it easy to change the vectors at runtime. The linker handles all this consistently. There is a device loader utility that checks and loads.

Notes: You must boot the system once. From then on, to change the driver, you write the new code and data onto the kernel using */dev/kmem*. There must be no changes in global data structures. Also, no device activity while reloading. You must understand static versus global data initialization — some things must be reinitialized when reloading, others are automatically reinitialized by reloading the data. The system is at risk — if you mess up badly, it can crash the system. You also must have writable executable code in the kernel.

OSx: Towards a Single UNIX System for Superminis

Ross A. Bott
Pyramid Technology, Inc.
1295 Charleston Rd.
P.O. Box 7295
Mountain View, CA 94039

OSx is a dual port of 4.2BSD and System V to a supermini by Pyramid Technology. It is not a layered system, but full implementation of both operating systems in a single machine.

The philosophy is that both 4.2BSD and System V programs should run under OSx without modification. Users should not suffer any performance penalties because of the coexistence of the operating systems.

The user interface is defined in terms of "universes." One may switch universes simply by giving the command *btl* or *ucb*. You can determine which universe you are in by the command *whereami*. Single commands may be executed in the opposite universe, and you may direct output between universes.

This system provides 4.2BSD and System V compatibility at a detailed level. There are no performance penalties and dual system transparency, essentially providing the best of both worlds.

New ½ Inch Tape Options and Trade-Offs for 4BSD on DEC VAX Processors

Robert J. Kridle
mt Xinu, Inc.
739 Allston Way
Berkeley, CA 94710

The performance of five tape units was measured in typical UNIX tape applications. They were benchmarked under 4.2BSD on a VAX 11/750. The units used were of two types: the traditional start-stop and the enhanced streaming technology tape sub-systems. The author does not recommend or disrecommend any of the units tested.

The traditional start-stop units tested were: Cipher 100X, Kennedy 9300, and NEC TD 1505. The enhanced streaming tape units tested were Cipher M891 and DEC TU-80. These systems cost anywhere from 5-20 thousand dollars.

The things measured included: average continuous data transfer rates (estimated and measured), and the measured time to complete the dump of a small file system. A recommendation for a modification to the 4BSD tape handler for the DEC TU-80 was made, which would improve its performance considerably.

Implementations II

January 20, 1984, 8:30 a.m.
Chair: Joseph S. D. Yao, Hadron, Inc.

Reporter: *Berry Kercheval*
Zehntel, Inc.
2625 Shadelands Dr.
Walnut Creek, CA

A Layered Implementation of the UNIX Kernel on the HP9000 Series 500 Computer

Jeff Lindberg
Hewlett-Packard
3404 E. Harmony Rd.
Fort Collins, CO 80525

Hewlett-Packard has implemented a UNIX kernel on top of an already existing operating system. HP-UX, as they call this implementation, is System III with the ever-popular "Berkeley enhancements," in this case the *ex/vi* screen editor. HP has also included *f77*, virtual memory, Pascal, memory shared between processes, 3-D graphics, and a data-base management system called "IMAGE."

The HP9000 Series 500 computers are based on a proprietary 32 bit microprocessor chip set with a stack architecture. It has user-transparent multi-CPU support, for which they included semaphores and a very carefully worked out scheduler.

HP-UX is based on top of an operating system called SUN (no relation to Sun Microsystems Inc.). The SUN operating system was intended to be a modern operating system to support HP's desk top BASIC system. As designed, though, it turns out to be language independent, and had several other features that made it attractive as a base upon which to build a UNIX implementation. It provided a high-level interface to the machine functions that need not be redone; was written in a high-level modular language [modcal — HP's modular Pascal language; I think it's proprietary but I'm not sure — it may merely be unsupported to customers] and is invisible to UNIX users.

There are four major sections to HP's UNIX kernel. First, memory management, which handles the user's private memory partition, virtual memory and shared memory partitions. Second, process management schedules processes to run and handles real-time stuff. Third, the file system, which needed to handle HP's diverse directory formats. Finally, the device drivers needed to be implemented so that new ones could be added dynamically.

The implementation strategy called for the implementors to layer their UNIX kernel on top of the SUN kernel. The aim was to implement exact System III syntax and semantics. Modcal was used instead of C. The HP-UX kernel was to run in privileged mode to allow access to the hardware and system data structures. Calls to SUN routines were to be made if appropriate. It was not a requirement that UNIX and BASIC co-exist.

The benefits of this approach were several. An existing high-level interface to he hardware would ease the initial "duh — how does this work" stage. Drivers for most devices already existed, and could be used without change. The IMAGE data-base intrinsic functions in the SUN kernel would make the UNIX implementation of IMAGE easy. SUN support for real-time functions, multiple CPUs, reliable error recovery, dynamic memory segments, semaphores and device I/O also made the "layered look" attractive.

Several risks also were apparent. What if they couldn't exactly duplicate the UNIX semantics? To solve this problem, HP developed a validation suite for the kernel that would verify its correctness. The other major worry was "how slow will it be?" Would there be too much translation from UNIX syntax to SUN syntax? As it turned out, performance was good, but tuning of the SUN kernel was necessary. Real time performance was also good.

In the process management section, the match between SUN and UNIX was good. The scheduler had static priorities and provided for system daemons. The major change was adding a *fork()* system call; the SUN kernel had no concept of "cloning" a process. Signals were easily integrated with the SUN trap handlers and periodic clock interrupts.

The filesystem provided the necessary hierarchical directories, but "special files" necessitated major changes to the SUN kernel. Things like *mount()*, *access()*, *pipe()*, and *chown()* all needed to be done from scratch.

The I/O system was a good match. The tty driver needed extensive modifications, though, and the internal keyboard and CRT drivers needed to be made to look like a terminal.

Memory management followed a simple model making the transition easy. The main work involved was in extensions to UNIX to support shared memory segments, mapping of files to memory segments, and virtual memory for large CAD applications — one of the major reasons for building the system in the first place.

Implementing the *exec()* system call was a significant effort, but rested on the SUN kernel operations. Shared libraries were implemented, but are available only for standard commands — they cannot be used by user programs. Other miscellaneous functions — automatic power-on boot, clock traps, multiple CPU support — were added.

Writing Device Drivers for Xenix Systems

Jean McNamara, Paresh Vaish, Richard N. Bryant
Intel Corp.
5200 NE Elam Young Pkwy.
Hillsboro, OR 97123

Mr. Bryant presented this "how-to" talk for novice writers of device drivers. Since Xenix is a direct UNIX port, these comments should be generally useful.

What a device driver does is manage input from and output to a hardware device. There are two types of device drivers, block and character. A block device driver is one that transfers entire blocks at once, such as raw disk or tape drives. Character devices are typically ones that communicate one character at a time, such as terminals, printers or plotters. In practice though, a character device is anything that isn't a block device. Block devices are generally simple. Most of the work is done by the basic I/O routines. Character devices can be quite complicated; the tty driver is a particularly baroque example.

All Xenix devices drivers are written in three files. A name for the device is invented, such as "tty" or "mt". "xxx" will be used in examples. The first file is called *xxx.h*. It contains the definitions of data structures, device addresses, control register masks, useful macros and other device-specific information. The configuration file, *cxxx.c*, contains the actual instances of the driver's private data. It lives in a "cfg" directory, and can be easily modified by customers with binary licenses who wish to reconfigure their system. The "cfg" directory is a Xenix-specific feature. Finally, the *xxx.c* file contains all the routines that do the actual work of the driver.

In a "cooked" block driver, nine routines are necessary. They were described briefly:

init()	initializes the device.
open()	opens the device, enabling device interrupts.
close()	closes it.
read()	handles requests to the driver to read a block from the device.
write()	handles requests to the driver to write a block to the device.
ioctl()	does device dependent magic things.
strategy()	receives I/O requests from, say read and write, queues them and starts I/O.
intr()	handles device interrupts like "I/O complete," and restarts the next I/O request.
start()	pokes the device to perform actual I/O transfers.

Raw block drivers are much simpler. *read()* and *write()* handle user requests for I/O, and do it, calling the system routine *physio()*. *ioctl()* does the same magic stuff.

Character devices are more complicated. *init()*, *open()*, *close()*, *read()*, *write()*, *ioctl()*, and *intr()* do similar things to the corresponding block device driver routines. *proc()*, however, is a System III/V requirement; it sets the "line discipline," required of all character devices.

l_open()	initializes the tty (or whatever) structure.
l_close()	clears it out.
l_write()	these two handle clist processing.
l_ioctl()	modifies the "tty" structure.
l_input()	places incoming characters into the input queue.
l_output()	takes characters from the output queue and sends them to the device.

In order to make a kernel with a new driver in the Xenix environment, the *xxx.c* file must be installed in */sys/io*, the *xxx.h* file in */sys/h*, and the *cxxx.c* file in */sys/cfg*. The makefiles must be altered to include the new driver, then a *make* in */sys/io* and a *makexenix* in */sys/conf* should do it.

First, though, there are four switch table that need to be updated. *initsw[]* lists the *init()* routines for all block devices. *intrsw[]* lists the interrupt routines. *cdevsw[]* lists *open()*, *close()*, *read()*, *write()*, and *ioctl()* for all character devices, and *bdevsw[]* lists *open()*, *close()* and *strategy()* for block devices.

Quite a few useful routines for the device driver writer exist in the kernel. *sleep()* and *wakeup()* allow process synchronization. Generally when a driver must wait for an event to happen it calls *sleep()*; when the event happens *intr()* calls *wakeup()* and the driver resumes processing.

Mutual exclusion is done with *spl0()-spl7()* and *splx()*, which set and restore the interrupt level. Finally *timeout()* will invoke a function some time in the future.

Mr. Bryant mentioned that the best way to write a new device driver is still to copy an existing one, but that this talk should help unfortunate souls without source licenses.

Porting Xenix to the Unmapped 8086

John Bruno Hare, Dean Thomas
The Santa Cruz Operation
Santa Cruz, CA 95060

At the beginning of his presentation, Mr. Thomas indicated that the paper would be published in *CommUNIXations* and that reprints would be available from The Santa Cruz Operation.

The Intel 8086 can operate in three modes, depending on what kind of memory management is done. The purpose of this presentation was to show that the "Small Model" 8086 is sufficient to support UNIX. Such machines as the IBM PC, Victor 9000, and Convergent Technologies' Integrated Work Station have no memory management system, and must perforce use this model.

The 8086 can access one megabyte of physical memory. Addresses are formed by combining a base pointer, which points to a segment on a 16K boundary, and a sixteen bit offset, which allows segments to be up to 64K long. There are four registers that can be used as base pointers called the "code," "data," "stack" and "extra" registers. Unfortunately, there are no provisions for intersegment protection or segment bounds checking.

In the "Small model" the code, data and stack segments are all the same. This corresponds roughly to the type 407 executables on PDP-11's. The "Compact model" has the text segment distinct from the data/stack segment, and corresponds to the "Separate I/D" executable on some PDP-11's.

The SCO Xenix port supports both of these models. They make relocation easy, as all the kernel needs to do is change the segment registers and move the code. The middle model, in which more than one text segment can exist, must trap intersegment calls so the kernel can change segment registers.

Two problems exist using the middle model, though. If there are any absolute addresses in the code of a program, they must be fixed up at load time. SCO includes a "fixup table" in the a.out module for this purpose. Another problem was swapping. If, say, absolute addresses are on the stack, and the processes gets swapped out and then swapped back in at a different address, it's in trouble. Either some means must be given to detect and fix these at swap-in time [ecch] or the process must be locked in core [ecch]. SCO's solution was to give the executing process high priority to reduce the likelihood of swapping, and they swap to the same physical address. Luckily most programs do not need to use the middle model; two which DO are *vi* and the RMS-Cobol Animator.

Interesting topics arose during questions. A malicious user can still trash the system, though. The kernel checks certain data structures for changes on each system call, and if they have changed, assumes the worst, issues a segmentation violation and kills the process. The large model is not supported, but is not written off yet either.

Transparent Implementation of Shared Libraries

Curtis B. Downing
Bunker Ramo Information Systems
35 Nutmeg Drive
Trumbull Industrial Park
Trumbull, CT 06609

Frank Farance
Systems Theory Design Corp.
555 Main Street, Suite 705
New York, NY 10044

Reporter: *Karen Summers-Horton*

This talk described a long needed feature in UNIX — a way to share large libraries among different running processes. It is not only useful to share big application libraries, but also the standard C library.

Shared libraries are defined as sharing a text segment among different processes. No attempt is made to share data segments. The traditional UNIX shared text mechanism (the **-n** option to *ld*) only allows sharing of the exact same text. Different programs loaded with the same libraries can't be shared. True shared libraries are better.

Various possible methods were discussed.

- Static links done by *ld*: the problem here is that if the code is changed, you must relink everything.
- Dynamic links (like MULTICS): you need hardware and software support.
- Quasi-static links: you must fake undefined symbols.
- Quasi-dynamic links: you must install stubs to shared library routines, and have the stubs dynamically link.

The method used is quasi-dynamic links. They do a lookup for the first call only, after this they must still go through the stub, but the overhead is very low.

The hardware and software required: standard *ld* and *cc*. You must have shared memory. You need a compiler with relocatable code, additional support to produce the stub, segmentation or paging memory mapping.

Things that belong in a shared library include *printf*, *scanf*, and applications support (in this case, a database management system). What does not belong in a shared library include system calls, low usage routines, and high performance routines.

One problem is that, since System V shared memory was used, the user must call a support routine to set it up, so this is not transparent to the user. Also, their compiler did not do enough relocation.

UNIX File Systems Optimization on Microcomputer Systems

David Robboy
Intel Corp.
5200 NE Elam Young Parkway
Hillsboro, OR 97123

This talk discussed different methods for making the UNIX filesystem faster. This has been a problem with UNIX for years, since the original filesystem was one of the major bottlenecks in UNIX. The object is to essentially "make your disk faster." This is even more important now since the new inexpensive winchester disks tend to be slow.

There are three possible bottlenecks in a filesystem. It can be I/O bound, limited by the transfer rate on disk (e.g. backups). It can be CPU bound, e.g. small data requests. It can be access bound, limited by the time to do seek or search operations.

Possible optimization strategies include getting faster hardware; using read ahead (this is good for sequential access in an access bound system, but bad for random access); buffer caching, to re-use data already read; and organizing the filesystem, for access bound systems, to reduce the scattering of files, and to reduce seeks.

They have already tried getting a faster controller, which increases the raw sequential I/O, but not multi-process throughput: real systems are usually access bound. They also tried a faster CPU, but it didn't help in proportion (only a factor of 1.5 or 2 for a CPU that's four times faster).

The 4.2BSD filesystem was considered. Berkeley claims a tenfold improvement in performance over the old 4.1BSD filesystem, due to a larger block size, clustering, and the rotational allocation of blocks.

Having a large block size is important. It gives the effect of read ahead, reduces scattering of data on the disk (because there are fewer, larger pieces). It also wastes space when small files are allocated, taking an entire block. Berkeley uses fragments, which avoid the wasted space, but are complex. They also cause fewer buffers in the cache (since each buffer must be larger), this is a disadvantage.

Cylinder groups partition the filesystem. This reduces file fragmentation, and arranges that files are closer to inodes and directories.

Rotational allocation may not be useful for multi-user random access. It introduces additional complexity, and must parameterize hardware dependencies.

There are other strategies. You can speed up memory intensive functions, such as memory to memory copies. You can improve buffer cache management, by adding more buffers, or having separate buffers for data and inodes. DMA data transfers can be used to go directly from disk into user space; this is good for program loading, and avoids clogging buffers with programs. You can maintain continuous files, which contributes to fast program loading. You can sort the disk free list.

The speaker concluded that UNIX is access bound, especially on micros. You can focus on keeping file blocks together. Some strategies enhance each other. Filesystems will always be bounded by either access, CPU, or I/O.

Communications

January 20, 1984, 10:30 a.m.
Chair: Mark Horton, AT&T Bell Laboratories

Reporter: *Berry Kercheval*

A UNIX to VAX/VMS Communications Link

Clifford N. Cary
Creare R&D Inc.
Box 71
Hanover, NH 03755

Creare has developed a system which enables a UNIX system to communicate with a VAX/VMS system. Many users can use the facilities for file transfer, remote terminals and virtual disk simultaneously. It was designed to support multiple error-free channels using layered software. It uses a typical UNIX command syntax.

A user process (or "transient" process as Mr. Cary termed them) communicates with a "net layer" background process. The "net layer" in turn communicates with a "data link" background process. The "data link" process talks to a device driver in the kernel which drives the communication line.

The "data link" process and the driver implement the DDCMP protocol over an RS-232 link. [I believe the standard tty driver is used, but I am not sure.]

The "net layer" acts as a multiplexer for multiple users, serializing user requests and ordering data coming from the other end of the link.

The system is unbalanced; that is, the UNIX system is the "master" and the VMS system is a "slave." [Let it be recorded here that the scribe resisted a strong temptation to editorialize.] Named pipes are used for data transfer from transient processes to the net layer, since they provide a "well-known" place to send data. Other architectural details of the implementation were discussed.

With the addition of a special device driver, "virtual disk" service can be provided. The UNIX transient process opens the device, which causes the kernel to spawn an instance of the disk server.

The server opens a channel to the VMS system, and from then on, a VMS file looks like a UNIX block device.

Mr. Cary claimed 400 character per second throughput on a 9600-baud RS-232 line.

Loving *uucp*, or How I Spent my Summer Vacation

P. Honeyman
Princeton University

D. A. Nowitz
AT&T Bell Laboratories
Murray Hill, NJ 07974

B. E. Redman
AT&T Bell Laboratories
Whippany, NJ 07981

Messrs. Honeyman, Nowitz and Redman spent their "summer vacation" rewriting the *uucp* software package, with the aid of a host of helpers — the list looks like a "Who's who on Usenet."

One change to "vanilla" *uucp* is the implementation of dynamically created per-site spool subdirectories. This greatly reduces the load of directory searching — a quadratic algorithm executed *in the kernel*. Also, since the sitename is in the name of the directory, sitenames can be fourteen characters, although this could cause problems for machines still running the older versions.

Logging and administrative information is much improved. Multiple *uuxqt*'s — one per site — are permitted, with the use of a **-s** flag to tell *uuxqt* to work only on a single site. Locking is improved. Instead of relying entirely on age to tell if a lock is stale, the pid is extracted from the lock file and signal 0 is sent to that process. By examining the error return from *signal()* it can be determined if the process still exists. This is possible only in USG UNIX, though the hack for 4.1 or 4.2 is easy — two lines in sys4.c.

Multiple L.sys entries finally work right. Many modems are supported. Other types of devices such as PBX's can be easily handled with "chat scripts."

cu has also been enhanced to use the same information about modems, switches, PBX's and ACU's that *uucp* uses.

Errors are returned to the originator. More tools are available to check on status of the *uucp* system or individual jobs, or to kill or reschedule jobs. Exponential back off on retry of failed connections has been added, and scheduling has been removed from *uucico*, and put in a separate program.

New Implementations of *uucp*

P. Honeyman
Princeton University

D. A. Nowitz
AT&T Bell Laboratories
Murray Hill, NJ

B. E. Redman
Central Services Organization

Security aspects of the new implementation of *uucp* introduced in the previous presentation were discussed here. Security objectives of the new design were to provide restrictive defaults, more flexible options, a more easily understood format for specifying the options, and to allow command permissions by site instead of compiling it into *uuxqt*.

The default permissions are obtained from the minimal permission file entry "LOGNAME somesite". This allows *somesite* to send files to the public directory, and execute *rnews* and *rmail*. Nothing else.

For *somesite* to receive files, "SENDFILES=YES" must be added to the permissions line. To allow *somesite* to request files, "REQUEST=yes" must be added. To allow *somesite* to write to other directories, "WRITE=dir1:dir2:dir3" is added.

An example of an "open system" would be "LOGNAME=uucpok REQUEST=yes READ=/ WRITE=/". Additionally, commands can be specified per site.

Future work will include encryption on spooled and transmitted data and gateway facilities to other networks.

[Mr. Nowitz later told me that this version of *uucp* was not a released product, and therefore not available outside of AT&T Bell Laboratories yet.]

Mapping the *uucp* Network

Rob Kolstad
CONVEX Computer Corporation
Dallas, TX

Karen Summers-Horton
2843 Valcour Court
Reynoldsburg, OH 43068

Reporter: *Jerry Deroo*

The speakers gave a quick overview of the size and growth of the *uucp* and Usenet network. A quick synopsis of the *uucp* network:

- more than 2100 sites
- sporadic connections
- low cost
- store/forward propagation
- some gateways to other networks
- a relative anarchy

Usenet, which is a subset of the *uucp* network, currently connects about 700 sites in North America, Europe and Australia.

To tame the beast, they are setting out to map the approximately 8000 *uucp* connections in a way which is usable by routing software. They are asking for public information concerning all sites. They anticipate that the project will take from six to twelve months to complete.

Panel Discussion

Following formal presentations was a panel discussion with Steve Bellovin, Peter Honeyman, Mark Horton, Rob Kolstad, D. Nowitz, Armando P. Stettner, Karen Summers-Horton, and Lauren Weinstein.

Participating from the audience were Rob Pike, Erik Fair, Andy Tannenbaum and many others. Several topics relating to the future of Usenet and the *uucp* net were covered with perhaps more heat than light.

Graphics

January 20, 1984, 1:30 p.m.
Chair: Noel Kropf, Columbia University

Reporter: *Jerry Deroo*

UNIX as a Development Base for High Performance Graphics Applications

Thomas Ferrin
Computer Graphics Laboratory
School of Pharmacy
University of California
San Francisco, CA 94143

This talk described work undertaken at UCSF to model molecular structures. Standard approaches to molecular modeling has resulted in models that were very cumbersome, and took a large amount of time and effort to create and change. An effort was undertaken with these objectives:

- allow the display and manipulation of multiple models simultaneously
- allow the selective display of a region of a model
- permit rotation and transformation of a model
- allow for the modification of a model (e.g., bond rotation)
- use color to enhance the representation of the model

To attain these goals, the following things were required of the supporting operating environment:

- an efficient and portable applications programming language
- ``real time'' response
- tools for optimization and documentation
- good programming examples to guide development

They chose UNIX in 1976, as it was a time sharing system to which people considered ``experts'' had already switched.

There were some disadvantages to this choice:

- no graphics hardware vendor support for the operating environment
- missing a decent Fortran compiler, which their users were accustomed to having.

They set out to write their own device driver and a graphics subroutine library. They made changes to the kernel to allow a real-time process to exist. The changes for real time process required about 40 lines of kernel code, and took about one month. They designed and implemented their own database and access methodology, optimized for protein and nucleic acid molecules.

The speaker then displayed a series of slides, and a motion picture of the system in action.

DAPS and GSDL: A Procedural Approach to Graphics

Christos Tountas
Graphic Information System Technology Inc.

This talk described the uses of a graphics system that has, as one of its interfaces, a procedural language to describe the graphics objects to display. Repetitive patterns can be generally described once, and then given parameters to indicate where the object is to be displayed.

The system supports many different output devices, and can present one of several different user interfaces.

There is presently a gap between a graphics package that has a collection of pre-cooked graphic objects that the user can chose from, and a package of graphics subroutines that must be enclosed with a complex user-written program before it can actually do anything. This package attempts to bridge this gap.

An example of the short-comings of most systems is that there is no way of obtaining automatic re-generation of an image after the image is changed: the entire image must be specified again. This suggests that parameterization would be a valuable feature: to enable the user to specify an attribute of the image in terms of other attributes.

The package supports the basic notion of move and draw, and supports a flavor of turtle graphics user-package interface. It is possible for a user to have a library of graphics objects from which other objects can be built.

The package operates in either two or three dimensions. It supports hidden line removal on a user controllable basis.

The speaker has an impressive slide show, demonstrating what the package can do.

The Design of a Dedicated Graphics Subsystem for a UNIX Machine

Or: How to Make Pictures in Real Time

Jack Burness
MASSCOMP
1 Technology Park
Westford, MA 01886

The speaker began by presenting a short history of the evolution of display systems. The path was:

- display lists: image drawn from memory in a "stroke refresh" manner.
  Problem: if display list was too long, flicker was likely.
- frame buffer: since operating system evolution makes display lists in user memory rather difficult, short of giving the display processor its own set of memory management registers (and this does not get by swapping), the display list is moved to the frame buffer.
- put the frame buffer into the terminal.

  These last two approaches make it difficult for the user process to get at the "display list"
- implement a ring buffer in user space that the display processor can access via the system bus. This is a hybrid approach, but works well if the display processor can lock user memory pages for the ring buffer; now the operating system can direct the display processor, since it is essentially doing DMA with it.

The speaker describes a system that is the result of this type of thought process. The system can support up to 15 virtual terminals on one display processor (15 is a magic number!). Each one is a full featured terminal, and is an overlay on the display processor. Each virtual terminal may have up to 15 viewports, or windows. (Again, that magic number.)

The real problem is one of data management: what to do with the data that is overlayed by another window? If you move it somewhere, it is possible to have to move 8 Mb of data. These moves have to be rather quick. And where to put it?

Image Processing Work Station

Dale K. Hensley
TRW
1 Space Park
Redondo Beach, CA 90278

This talk described the evolution of a software package for performing image processing.

The software has undergone a lot of change. As the software was originally environment specific, each move caused a re-write. They finally moved to UNIX, and have stuck to basically V7 system calls, to enhance the portability. Despite their intentions, however, every once in a while device dependence crept into the software. Things like the size of an integer would be hardwired in every once in a while.

Multiple copies of #include files get you in the end, as well.

They ran into one problem, in that no one had ever written anything on how to write a device driver. This results in device drivers for exotic graphics boards that look like line printer drivers, if one is not careful.

Their current configuration is a VAX 780 with three Deanza image processing workstations. Each of the Deanza devices has two Unibuses, so they use one to connect the device to the VAX, and hang an LSI-11/23, through a Qbus/Unibus converter, off the other. The 11/23 runs the 2.8BSD kernel, which buys them a few performance improvements:

- the 1K block filesystem improves throughput
- expanded memory capacity of the 23+ lets them run 2 Mb of memory, gives them a memory disk
- they use memory maps for large copyins/copyouts. This speeds up image manipulation and polygon filling.
- The VAX can access the 11/23 memory to do DMA.

They are now planning to develop a product based on this work, to deliver a low cost image analysis system. The system would cost between 30 and 40,000 dollars, and be made up of:

- LSI-11 (23+ or 73)
- tape drive
- 512 by 512 monitor
- array processor

One aspect of the implementation is that the user is not exposed directly to the UNIX environment running on the 11/23. This means that the users do not require any UNIX knowledge.

This talk was pressed for time as well.

Real-Time UNIX

January 20, 1984, 3:30 p.m.
Chair: Reidar Bornholdt, Columbia University

Reporter: *Berry Kercheval*

Real time Extensions to the UNIX Operating System

Brian Look, Gary Ho
Hewlett-Packard
Data Systems Division
11000 Wolfe Rd.
Cupertino, CA 95014

A real-time system reacts predictably and in a timely manner to events in the "real" world. Applications for real-time systems include process control, machine monitoring, instrumentation or laboratory automation.

The primary problem in integrating real-time processing into the UNIX environment is in management of the real-time process. Basically, the running process must be pre-empted if a higher priority process becomes ready to run, say, in response to a device interrupt that must be serviced NOW, not in fifty milliseconds. Thus the need is for priority based pre-emptive scheduling in which the priorities do not degrade over time. Round-robin scheduling of real-time processes is necessary, not time-slicing, and it must be possible to control the priorities of the real-time processes.

The scheduler must have micro-second resolution, and not drift over time. Dispatching of processes must be fast; the system often cannot wait for a long system call to complete, so system calls must be interruptible.

Shared memory is often useful. Either Bell's segments or Berkeley's mapped files will do. Some things need to be locked in memory: shared segments, memory-mapped files and perhaps very high priority process images.

Process synchronization presents some interesting problems. A guaranteed maximum interrupt response time is necessary, as are reliable signals, shared memory and semaphores. The isolation of critical regions into areas protected by semaphores enables higher priority processes to be dispatched without worrying about whether they will destroy critical data structures.

File system modifications may also be needed. Pre-allocation of disk space would save critical time in servicing high-priority processes that need to write to the disk. Device and file locking is already provided in 4.2. User control of buffering — better than just calling *sync()* — would be nice, as would contiguous files to ensure faster startup. Scatter read/gather write was also mentioned as desirable.

For most real-time applications, user-written device drivers are inevitable, but folks stuck with binary-only licenses are out of luck here. Non blocking I/O with an "I/O complete" signal is attractive in this context. Synchronous I/O multiplexing, like *select()* in 4.2, and preservation of I/O sequencing is important.

Real-time privilege control can be implemented with two kinds of user. Ordinary users are normal users, while real-time users can invoke processes with some or all of these real-time features. Two possible mechanisms for invoking real-time features would be special files that automatically get the features when loaded, and *ioctl()* calls to invoke them.

In summary, the presenters feel that a combination of the best features of 4.2BSD and System V, with real-time priorities, memory locking and user-written drivers will ensure good real-time performance.

UNIX Performance Optimization — UNIX/68000/SKYFFP

Joseph F. Germann
Sky Computers, Inc.
Lowell, MA

The SKY floating point processor can increase performance of both floating-point and integer functions when integrated into a 68000 system. One might think that with all the work done on the kernel over the years, it would be pretty tight, and there would not be much room for improvement. One might be wrong.

The SKYFFP is a microprogrammable 2901 bit-slice floating point processor on a single Multibus Board that can perform a 16×16 bit multiply in 120 nanoseconds — about the time to access some fast memory chips! It runs as a slave coprocessor to the 68000, and has six registers in the I/O page for command, status, microcode and data. It will do complex, trigonometric, logarithmic and long integer — signed and unsigned — operations. The device driver for this beast is not a normal driver. For one thing, it can support multiple users at once.

The basic paradigm for using the SKYFFP is to load the command, load the data, and then read the result when it is ready. An add will thus take a minimum of four 68000 instructions. The SKYFFP will then hold the bus signal DTACK until the operation is done.

Usually floating point arithmetic is done with runtime routines. The arguments must then be saved on the stack, the routine must be called, it does its thing, then saves the result and returns. A typical floating point add may take from 70-150 microseconds, with 18 microseconds for procedure call overhead. The SKYFFP is integrated into the C language using a mix of in-line code generation and run-time libraries. Fast operations are coded in-line (by a special compiler) to avoid the procedure call overhead. They include add, mul, sub, cvt, and integer ldiv and lmul. Longer functions — log and trig, etc. — are coded in runtime libraries as the overhead penalty is not as high a fraction of the compute time. A 32 bit IEEE FP division can be done in 25 microseconds, for example, while a tangent may take 180 microseconds.

So what's the bottom line? In a Codata 68000 system with the SKYFFP, and five of the most commonly used kernel math operations on long integers microcoded into the SKYFFP, compiles were 100% faster, I/O intensive processes were five times faster, and memory intensive processes were 40% faster. (Measured with the Teus Hagen/Andrew S. Tannenbaum benchmark). The naked 68000 ran 70,000 Whetstones/second, and with the SKYFFP it jumped to 294,000 Whetstones/sec.

Life with UNIX in Real Time

Steven T. Polya, Jeff Barnes
Contel Information Systems

Messrs. Polya and Barnes presented a case study of Contels implementation of a real-time network control system on a Codata 300 68000 system with UNIX. They had a binary license, but could write device drivers. What they felt they needed was good inter-process communication, a commonly addressable memory resident database, priority based scheduling and non-interruptible processes.

A Z80 based network controller neeeded to interface with the 68000 UNIX processor. This was done by using a 128K ram card mapped in with *phys()*. Two FIFO ring buffers were implemented for interprocessor message storage.

Since many processes use net services, IPC was also implemented using ring buffers in the leftover RAM on the card. This enabled them to do away with pipes and signals for this service, as realtime processes should never sleep. Access control for these data structures was added to the kernel.

An interesting dynamic priority assessing scheme was presented.

An event pseudo device driver was implemented to enable a process to sleep on an event. A master process finds the events and signals the sleeping process to awaken.

To eliminate swapping problems, lots of RAM was installed, and processes could be locked into it.

Integrating a Peripheral Floating Point Processor in UNIX

Eryk Vershen
UniSoft Systems
739 Allston Way
Berkeley, CA 94710

Mr. Vershen presented a brief outline of his integration of the SKYFFP into a UNIX system. This board was described in a previous talk.

Maximum speed was desired, so system calls to the SKYFFP were considered too slow, but using *phys()* to access the SKYFFP registers was very MMU dependent, and besides only the super-user can do it. A hook was added to the SKYFFP device driver to allow it to perform the *phys()* for the client process. This enables a malicious user to zap the microcode, but TAANSTAFL.

The system call overhead in the UniSoft kernel was 250 to 400 microseconds. Using simulated floating point, additions took 270 microseconds, so clearly the system call approach was too slow. Inline code gave the best performance, reducing add times to 80 microseconds.

Information on Vendor Exhibits at UniForum

Each of the exhibitors at the Washington DC UniForum Conference was invited to submit 100 words describing their products or services for inclusion in this issue of *;login:*. All material presented has been reprinted below with only minor editorial changes.

Able Computer

Communication solutions for DEC and IBM systems. Before installing networks, switches, multiplexors or other communication systems as devices Able may provide a more cost effective and longer range solution. Able's selection of software transparent solutions are the most advanced and cost effective in the industry. If your problem is connectivity — see Able.

For further information contact:
Bob Nicholson
Able Computer
1732 Reynolds
Irvine, CA 92714
714-979-7030

Absolut Software

Absolut offers a line of complete inventory management and business systems for chainstore retailers, mailorder houses, distributor/wholesalers, and manufacturers. The software is written in C and incorporates the Unify data base and Report Generator and will run on any 68000 or larger computer.

For further information contact:
John T. McGrath
Absolut Software
2001 Beacon Street
Boston, MA 02146
800-ABS-UNIX, 617-277-0610

Amdahl Corporation

Amdahl's UTS (Universal Timesharing System) combines the functions of UNIX with the power of a large mainframe. UTS runs on any S/370 architecture processor, including both Amdahl and IBM CPU's.

For further information contact:
Cynthia Ainsworth
Amdahl Corporation
PO Box 3470
Sunnyvale, CA 94088-3470
408-746-8945

Apollo Computer Inc.

Apollo Computer Demonstrated AUX, the firm's adaptation of UNIX System III software with Berkeley extensions. Operating in a multi-node local-area network, AUX allows users full Bell and Berkeley UNIX functionality in addition to Apollo's standard windowing, networking, and graphics capabilities.

The actual hardware consisted of two DN300 monochromatic and one DN600 color, computation nodes. One of the DN300's was running "diskless," thus showing Apollo's unique capability of supporting virtual memory, via demand paging, over a local area, 12 megabit per second, network.

For further information contact:
John Bowne Jr.
Apollo Computer Inc.
15 Elizabeth Drive
Chelmsford, Mass 01824
617-256-6600

Applied Data Systems, Inc.

Applied Data Systems, Inc. exhibited the Vector Scan 512 color graphic controller that sells for $975.00 at quantity one. This unit provides 512×480 pixel resolution at 4 bits per pixel. Both monochrome and color monitors are supported by the Vector Scan 512. Also, hard copies of the graphic images can be obtained on various low cost dot matrix printers.

For further information contact:
Robert Kondner
Applied Data Systems, Inc.
9811 Mallard Drive, Suite 213
Laurel, MD 20708
301-953-9326

AT&T Technologies

Products exhibited:

- UNIX System V (release 2.0) enhancements include a 5 to 10% improvement in overall performance. More than 500 fixes, new and enhanced commands, new utilities such as job control, and improved documentation.
- UNIX Documenter's Workbench* Software — complete text processing with device independent *troff*.
- UNIX Writers Workbench* Software — more than 25 powerful programs for all types of writing.
- UNIX Instructional Workbench* Software
- UNIX System V support and training.

New products announced:

UNIX System V (Release 2.0)
UNIX Documenter's Workbench Software
Basic Interpreter
NC 68000 Software Generation Systems

---

* Trademark of AT&T Technologies

Other Announcements: AT&T Technologies and Digital Research Inc. announce joint development of an applications library for UNIX System V. Library [?] will evaluate and market UNIX System V applications from independent software vendors (ISV's).

For further information contact:
O. L. Wilson
AT&T Technologies
PO Box 25000
Greensboro, NC 27420
1-800-828-UNIX

Avalon Computer Systems, Inc.

Avalon demonstrates it's VAX attached processor system. This system is available now as an addition to VAX Systems. Adding one or more Avalon systems dramatically increases the computing power of a timesharing system. Each Avalon AP/10 processor includes one million bytes of memory and also includes floating point hardware. The AP/10 is completely contained on one Unibus circuit board and provides computing power similar to the 11/750. Avalon supplies the compute-bound UNIX utilities such as *nroff*, *troff*, and the C compiler. Applications may be run on the Avalon system after a simple recompilation of the source.

For further information contact:
Conrad Schneiker
Avalon Computer Systems, Inc.
55 North St. John Avenue
Pasadena, CA 91103
818-796-0861

AVIV Corp.

AVIV Corp. designs, manufactures and markets high performance controllers and subsystems for DEC, DGC and Multibus Based computers. Tape controllers are compatible with all leading models of industry standard streaming and start/stop drives. In addition, AVIV has the broadest range of GCR (6250 bpi) compatible peripherals, including CDC, STC and Telex Drives. AVIV disk controllers are available with standard SMD drivers from 80-675 Mb, and with fixed or removable media. The Unibus and VAX 11/750 CMI bus controllers also support advanced SMD drives, such as the CDC 9715-500 and the Fujitsu Eagle, which have a 1.8 Mb/sec transfer rate. AVIV controllers are compatible with all DEC and DGC operating systems, as well as UNIX and UNIX derivative operating systems.

For further information contact:
Ed Arsenault
AVIV Corp.
26 Cummings Park
Woburn, MA 01801
617-933-1165

Benjamin Cummings Publishing Co.

Benjamin Cummings Publishes an outstanding list of UNIX, Xenix, System V and C books. The following titles are now available:

A Practical Guide to UNIX by M. Sobell — Published 1984
A Book on C by Kelley and Pohl — 4/84
A Practical Guide to Xenix by Sobell — 6/84
A Practical Guide to System V by Sobell — 7/84

For further information contact:
Jake Warde
Benjamin Cummings Publishing Company
2727 Sand Hill Road
Menlo Park, CA 94025
800-227-1936

Britton-Lee, Inc.

Britton-Lee's IDM (Intelligent Database Machine) is a Relational Database Manager implemented in a dedicated function processor. The IDM achieves high performance (5-10 times faster than software implemented Relational DBMS) by offloading the DBMS to special-purpose "Backend" hardware. The IDM can function as a shared data server for two or more different host CPU's. Supported hosts include:

IBM PC/PC DOS
VAX/UNIX (Berkeley 4.2BSD)
PDP-11/UNIX (Version 7.0)
VAX/VMS

Host support software provides interactive keyboard queries, queries embedded in applications programs and extensive subroutine library extensions.

For further information contact:
Kathryn Scollon
Britton-Lee, Inc.
14600 Winchester Boulevard
Los Gatos, CA 95030
408-378-7000

BRS

BRS/Search is a text storage and retrieval system that permits database creation, loading, maintenance, and full-text/free-text searching. The systems currently operates on hardware from seven companies under the UNIX operating system. Applications include litigation files, office correspondence, engineering records and documents, personnel files, library files, and many others. BRS/Search permits retrieval on any word, parts of words, and combinations using Boolean, positional, and relational operators. There is no limit to the length of records or fields. The system has been used to store the full text of textbooks and encyclopedies. Dealer and OEM inquiries are invited.

For further information contact:
Richard C. Simon
BRS
1200 Route 7
Latham, NY 12110
518-783-1161

Cadmus Computer Systems, Inc.

Cadmus 9000 systems are 32-bit multi-purpose computers that deliver the power of a mainframe over a distributed network environment. Designed from the bottom up as a networked node, each Cadmus system provides users with a dedicated MC68010 microprocessor, an extensively enhanced multi-user UNIX Operating System, Q-Bus and Multibus peripheral compatibility, virtual memory and high resolution color graphics.

Available in multiple configurations, ranging from diskless workstations to high capacity multi-user networks, Cadmus systems are networked over Ethernet or 50 Mbit/sec Fiber optic Net, into a shared-resource system environment of unprecedented capabilities. UNISON, the Cadmus distributed network software, makes full, transparent access to the entire networks processing resources available to each individual node.

For further information contact:
David Schell
Cadmus Computer Systems, Inc.
600 Suffolk Street
Lowell, MA 01854
617-453-2899

Callan Data Systems

Callan Data Systems demonstrated the UNISTAR family of Multibus super microcomputers running UNIX System V. The UNISTAR 100 single user and UNISTAR 200 multi-user are 8-slot 68000 based systems with up to 43 Mbyte disks. The UNISTAR 300 is a 12-slot 68010 10 MHz system with 2 Mbytes RAM, 172 Mbyte disk and 45 Mbyte tape backup. Languages supported include C Compiler, Fortran 77, Pascal, Ada, Cobol and Basic. Options include floating point accelerator, array processor, intelligent controller, TCP/IP protocol and networking. Optional application software offered are Easy-Type word processing, UniCalc spreadsheet, and MicroIngres relational database management system.

For further information contact:
Bob Hufnagel
Callan Data Systems
2645 Towns Gate Road
Westlake Village, CA 91361
818-991-9156

CGA Computer Inc.

CGA is one of this country's largest full service data processing consulting firms. We are a fifteen year old firm based in Holmdahl, NJ and employ 600 full time consultants nationwide in 17 offices in 14 states. One of our largest clients is AT&T Bell Laboratories, where we employ 80 UNIX and C professionals. We are a publically held OTC (NASOAQ) with sales projected at $40 million for the current fiscal year. CGA offers a wide variety of DP related services from feasibility studies to systems design and analysis to contract programming and technical skills training. CGA most recently entered the software marketplace with many products for the IBM environment.

For further information contact:
Tom Drew
CGA Computer Inc.
1011 E. Touhy Avenue
Des Plaines, IL 60018
312-296-9660

Cambridge Digital Systems

Cambridge Digital Systems is an integrator of customized DEC, NCR and UNIX compatible systems and software. The company's UniForum exhibit focused on it's new 68000/68010 based Univax CPU, designed to optimize UNIX performance. On display was the Univax configuration of CDS' prepackaged Q-bus based System 94 line, running a true adaptation of System III, System V and Berkeley 4.2 UNIX with a 70 Mbyte Winchester and ¼″ streaming tape. The Univax System 94 is a compact twelve user system with a 10½-inch desk top, rack-, or floor-mountable enclosure. The East Coast distributor of the NCR Tower 1632, CDS exhibited this sixteen user system running an adaptation of UNIX System III.

For further information contact:
Dena Wortzel
Cambridge Digital Systems
PO Box 568, 65 Bent Street
Cambridge, MA 02139
617-491-2700, 800-343-5504

Charles River Data Systems

Charles River offers the Universe 68 line of OEM Super Micro computers — designed for high performance and low cost (over 1 MIPS for under $10,000). Running UNIX System V or the UNOS operating system with its real time and high reliability extensions. Configurations range from 10 Mb disk with 512 Kb RAM and 4 users to 5+ Mb RAM, 200+ Mb disk and up to 64 serial ports. The systems are built on 12.5 MHz 68000 main processors with cache and a second 68000 processor for character device control. Floating point, array processors and graphics interfaces are also available.

For further information contact:
Jim Isaak
Charles River Data Systems
983 Concord Street
Framingham, MA 01701
617-626-1000

Cifer Inc. / C&P International Consultants Inc.

Cifer, a British manufacturer of terminals and microcomputers which feature advanced multiprocessor architecture and high resolution graphics, introduces it's new UNIX product line of Personal Workstations, Personal Microcomputers, and Multi-User Microcomputers. Cifer UNIX machines contain the powerful 16-bit MC68000 with UniSoft's UniPlus version of Bell Laboratories System III UNIX operating System. All have built-in Winchester, 800 Kb floppy disk, RS-232 serial interfaces, Centronics parallel interfaces, IEEE-488 bus, external floppy disk bus, and a Z80A 128k processor with CP/M Plus operating System. All Cifer Workstations contain an intelligent VDU on a separate Z80A processor and provide a high selection of transmission and display functions.

For further information contact:
Gene Bidwell
Cifer Inc.
375 N. Broadway, Suite 302
Jericho, NY 11753
516-935-8490

CMI Corp.

CMI Corp. is displaying an IBM Series 1 running under SERIX, a full implementation of System III UNIX. Products complimenting SERIX are an enhanced C compiler, a visual editor, a C compiler under EDX, RM Cobol, the Unify data base and ViewComp Spreadsheet accounting. Products soon to be available are an EDL to C converter and a word processing package.

For further information contact:
Dennis J. Pennington
CMI Corp.
2600 South Telegraph
Bloomfield, MI 48013
313-362-1000

Compugraphic Corp.

We displayed a Compugraphic 8400 digital typesetter being driven directly be a modified *troff* program prepared by Technical Type and Composition of Salem, OR. During the show we ran this *troff* program on a Pixel; however, this software is portable to most UNIX based systems. The use of this software to drive the 8400 will allow a company to directly typeset their documentation. Compugraphic Corp. is a supplier of typesetting equipment in the low to medium price ranges and is making an effort to be as compatible as possible with *troff* and UNIX based systems.

For further information contact:
Carl E. Miller
Compugraphic Corp.
200 Ballardvale Street
Wilmington, MA 01887
617-658-5600, x5087

Computer Consoles, Inc.

Officepower^{tm} is a multifunctional, highly integrated office automation system providing turnkey applications in support of general office automation functions, plus a relational database mangement system, a user-defined forms package, a query language and a report generator.

Officepower runs on the Power 5/20 standalone, one of CCI's family of MC68000 based computers produced by CCI. The Power 5/20 can be expanded into multiprocessor systems that support a network of terminals and are designed to incorporate CCI's fault-tolerant "Perpetual Processing^{tm}" system.

For further information contact:
Edward W. Scott Jr.
Computer Consoles, Inc.
97 Humboldt Street
Rochester, NY 14609
1-800-833-7477

Computer Services Corp.

Terranet is an inexpensive local area network. Tap units use RS-232C to connect to various forms of data equipment (e.g. terminals, modems, printers, and computers). Data is sent to other tap units along a coaxial cable which can extend up to 500 meters. Any tap can connect to many machines without running many cables.

Time division multiplexing is used to provide 30 9600 baud, non-contending, virtual circuits. Switch selectable firmware within reach of each tap can provide a conversational dialogue at user terminals, short status codes at computer connections or gateway actions when taps are used to connect two networks.

For further information contact:
Mr. M. Fujita
2-6 Nishi Shinjuku, Shinjuku
Tokyo, Japan 160-91
201-569-6300

Computer Technology Group Telemedia, Inc.

The computer Technology Group specializes in UNIX and C training given in public and on-site seminars, as well as offering video based training and interactive videodisk training courses. CTG's video based training integrated professionally developed and produced video and text materials, including hands-on exercises. Our courses are produced with the highest standard of video quality applying the latest technique of instruction design including the use of computer graphics and animation, compressing learning time.

CTG's interactive videodisk system, co-developed with Interactive Training Systems, Inc. contains computer generated text and graphics designed to direct students and monitor their progress through CTG's video training materials.

For further information contact:
Gary Slavin
Computer Technology Group Telemedia, Inc.
310 S. Michigan Avenue
Chicago, IL 60604
312-987-4082

Computerized Office Services Inc. (COSI)

COSI is a full-serviced UNIX software house dedicated to serving the commercial data processing industry. At UniForum '84, we introduced two new product lines:

- VISUAL — visually-oriented product development tools for commercial data processing and
- LINK — communications packages linking personal computers, UNIX machines, and mainframes.

COSI also provides UNIX software services such as:

- system design of complex UNIX-based corporate information systems,
- conversion of software, both custom and product, to a UNIX environment, and
- educational assistance for DP staffs adapting to a UNIX culture.

For further information contact:
Keith Burleigh
Computerized Office Services Inc.
313 North First Street

Ann Arbor, MI 48103
313-665-8778

Computerware, Inc.

Computerware, Inc. distributes a full range of computer supplies, furniture and peripheral equipment. They are also the authorized Virginia dealer for the Texas Instruments Professional Computer. Featured at the UniForum conference were the Ambassador and Guru video display terminals manufactured by Ann Arbor Terminals, Inc. These multi-featured terminals are favorites among UNIX users.

For further information contact:
John Dargo
Computerware, Inc.
8480-I Tyco Road
Vienna, VA 22180

Conetic Systems, Inc.

Conetic Systems markets a data base and application generation system written in C and fully integrated accounting modules developed with the application generation system. The product runs identically under UNIX and MS/DOS. Multiple utilities have been written to support virtually all types of data management including relationships. All data types and file structures including hashed, ISAM, and sequential files. Files can be loaded to spread sheets or other programs or accepted from other data bases. The product is presently used by Fortune 500 companies, including NCR and Allied Corp., with more general distribution now in progress. For information contact the above.

For further information contact:
R. Richard Schwindt
Conetic Systems, Inc.
12025 N.E. Marx, Suite 5H
Portland, OR 97220
503-253-0074

Cucumber Bookshop, Inc.

UNIX and C language book display including all books currently available, and catalog of forthcoming books due in the next 3 to 6 months; the most complete list of UNIX books so far as we know.

Also, demonstrating an information retrieval system, SIRE* running on UNIX, with automatic full-text indexing, ranked output of usefulness, word stem matching, related term searching, variable length records, and more.

For further information contact:
Dave Harris
Cucumber Bookshop, Inc.
5611 Kraft Drive
Rockville, MD 20852
301-881-2722

---

* Trademark of KNM, Inc.

Data General Corp.

Data General's UNX/VS is UNIX operating system environment for the 32-bit Eclipse MV/family systems. It is fully integrated with DG's AOS/VS operating system thereby providing a dual environment for the user. All AOS/VS applications, utilities, languages, debuggers and communication facilities may be accessed from UNX/VS.

The UNX/VS operating system environment is based on AT&T's System V UNIX and includes Berkeley enhancements, i.e. C shell, *vi* and *termcap*.

For further information contact:
Barbara Loonam
Data General Corp.
4400 Computer Drive F-134
WestBord, MA 01580
617-366-8911 x5281

Dual Systems Control Corp.

Dual produces highly-reliable, exceptionally high-performance supermicro systems that meet the industrial standards of professional engineers. Dual first offered the MC68000 with UNIX in 1981 and has established a solid reputation for delivering stable, proven products. Dual delivers several major technical advantages over competing 68000 based systems including DMA on serial RS-232C outputs, 256b FIFO buffer on serial inputs, and an SMD disk controller that supports up to 1.2 Gb of Winchester disk and provides the highest throughput on a UNIX system. Also available on these 12- to 16-user systems are floating point processing, up to 3.25 Mb of RAM, 9-track tape drive and interface, graphics boards, Fortran 77, Pascal, RM/Cobol, Basic, Lisp, Real Time Kernel, Lex word processing, MBSI accounting, ViewComp spreadsheet, MicroIngres, Unify, and, of course, System V with Berkeley enhancements.

For further information contact:
Joan R. Stivers
Dual Systems Control Corporation
2530 San Pablo Avenue
Berkeley, CA 94720
415-549-3854

ELXSI

The ELXSI System 6400 is a true 64-bit multi-processor supercomputer running System V. It consists of a 320 Mbyte/second I/O capacity. CPU's are built using proprietary ECL chips and run 4 million Whetstone instructions/second EACH, for system speeds from 4 to 40 MIPS. Processes each have up to 4 Gb of virtual address space. Languages: C, Pascal, Fortran-77, Basic, Cobol-74, Main Sail. High performance Ingres-based relational database. Outstanding price/performance ratio.

For further information contact:
Dave Reed
ELXSI
2334 Lundy Place
San Jose, CA 94131
408-942-0900

Excelan, Inc.

"UNIX on Ethernet^{tm} with computer inter-operability" was the theme of Excelan's booth at the 1984 UniForum Conference. The Excelan Open Systems (EXOS^{tm}) concept was demonstrated with the integration of four difference UNIX-based computers, each from a different manufacturer designed for applications from general business to factory automation and CAD/CAM, each sharing the resources of the other. This was made possible by Excelan's EXOS 101 Ethernet front-end processor boards and on-board EXOS 8010 TCP/IP protocol software which are independent of host CPU type, UNIX version or source of the port. Also shown was the Excelan Nutcracker^{tm}, the world's only analyzer/simulator for Ethernet systems.

For further information contact:
Doug Planchon
Excelan, Inc.
2180 Fortune Drive
San Jose, CA 95131
Telex 176610
408-945-9526

Federal Marketing Group, Inc.

Federal Marketing Group, Inx. is a distributor offering rapid availability of UNIX- and MS/DOS-based Hardware and Software at aggressive prices coupled with strong systems engineering support. Inquiries are invited from dealers, systems houses, government agencies, educational institutions, and large corporations. Lines offered include Altos 68000, 586, and 986; APPGEN program generator and accounting software from Software Express; R Word; Horizon integrated office automation; Texas Instruments professional and portable PC; Avatar PC/3278 computer and terminal; Samna word processing; Executec SeriesOne; Mosaic Integrated 6 with graphics; and other superior products. Dealers are eligible for free training in UNIX!

For further information contact:
David E. Fedder or Elizabeth A. Fedder
Federal Marketing Group, Inc.
100 West Hill Street
Baltimore, Maryland 21230
301-539-2121

Fortune Systems

Fortune Systems has over 30,000 workstations worldwide, currently the largest installed base of UNIX systems. Fortune Systems' multiple user micro computer is designed for both small and large business with such applications as: word processing, spread sheet, business accounting applications, Fortran, Cobol, C, and Basic, plus a wide variety of third party software. Fortune offers complete support, training and maintenance programs.

For further information contact:
Carolyn Carnifix
Fortune Systems
101 Twin Dolphin Drive
Redwood City, CA 94065
415-595-8444

Government Computer News

Government Computer News is a monthly publication dedicated to the interests and needs of computer managers and professionals, and end-users of computer products and services throughout the government.

For further information contact:
David Ross
Government Computer News
1620 Elton Road
Silver Springs, MD 20903
301-445-4405

Handle Corp.

The Handle Family of products is a state-of-the-art, interactive line of software for office automation, which runs under the UNIX operating system. All Handle products are completely function key driven and offer such unique features as: archiving, proof mode, and in-text graphics. The Handle Family of products includes: Handle Writer (word processor), Handle Spell, Handle Graphics, Handle Calc, Handle List (database management system), and Handle Access (access to mainframe database).

For further information contact:
Dianne Harlow
Handle Corp.
PO Box 7018
Tahoe City, CA 95730
916-583-7283

Hayden Book Co.

Hayden Book Company has begun distribution of AT&T Bell Labs UNIX system videotapes. They explain and discuss the UNIX operating system, the most portable and expandable operating system in today's computer environment. The tapes provide visual explanation and discussion of the UNIX System V, it's concept, operation and applications. Explanation is provided by the creators and pioneers of the UNIX system, including Kenneth Thompson, Dennis Ritchie, and Brian Kernighan. The tapes, produced by Belove-Laiserin, are available in both Beta and VHS formats.

For further information contact:
Barbara C. Garris
Hayden Book Co.
10 Mulholland Drive
Hasbrouck Heights, NJ 07604
201-393-6000

Hewlett-Packard Corp.

HP offers a range of high-quality, high performance systems which run HP-UX, HP's enhanced version of System III UNIX. Products featured at the show included engineering workstations and multi-user systems. All systems highlighted graphics capabilities.

For further information contact:
Doug Hartman
Hewlett-Packard Corp.
19447 Pruneridge Ave.
Cupertino, CA 95014

408-725-8111

Human Computing Resources

Human Computing Resources exhibited Chronicle[†], its new line of advanced business applications software especially created for the UNIX operating system. Chronicle is particularly well suited to 16 and 32-bit microprocessor systems. It may be run on almost any machine using UNIX and is portable to many major relational databases. As well, HCR displayed its UNITY[†] operating system for a NS16032-based machine and VAX UNITY under VMS[‡], its emulation of UNIX giving VMS users access to both systems in a correct and non-privileged manner.

For further information contact:
Jim Peters
Human Computing Resources
10 St. Mary Street
Toronto, Ontario M4Y 1P9 Canada
416-922-1937

Integrated Business Computers

IBC displayed the IBC Ensign super-micro. The Ensign is an extremely fast multi-user UNIX System III system. The Ensign supports 32 users with up to 8 Mbyte of memory and over 1000 Mbyte disk storage. In late first quarter the Ensign will support UNIX System V with the MC68010. The Ensign was designed to run multi-user UNIX faster than any other super-micro.

For further information contact:
Randy Rogers
Integrated Business Computers
21621 Nrodhoff St.
Chatsworth, CA 91311
818-882-9007

Information Industries, Inc.

Information Industries, Inc. (Triple-I) provides a flexible, risk-free alternative to traditional hiring and recruiting methods. This alternative, called Skills Management, allows Fortune 500 companies to overcome problems like head count limitations, peak load demands, and high recruiting costs. The result is increased productivity and cost-efficiency.

Triple-I's Skills Management Program delivers these benefits by providing highly skilled data processing/engineering consulting professionals who are experts in areas like networking, UNIX, C, M-204, OS/MVS, to name a few. Through constant exposure to state-of-the-art environments and career planning techniques, these Triple-I professionals are able to greatly enhance their career opportunities and potential.

For further information contact:
David Francis
Information Industries, Inc.
8880 Ward Parkway
Kansas City, MO 64114
1-800-triplei, 816-444-8100 in Missouri

[†] Chronicle and UNITY are trademarks of HCR.

[‡] VMS is a trademark of Digital Equipment Corp.

Integrated Computer Systems

Integrated Computer Systems is devoted to providing unbiased practical education in advanced technology. The courses provide engineers, system designers and their managers with a solid technical background and up-to-date information on the state-of-the-art in applying technology. Formats include 4-day short courses, a self-study program, video courses and custom designed on-sites. Subjects include Networks & Systems, Man/Machine Systems, Digital Processing, Minis & Micros, Software and Management courses.

For further information contact:
Carolyn Yost
Integrated Computer Systems
6305 Arizona Place
Los Angeles, CA 90045
213-417-8888

Interlan, Inc.

Interlan, Inc. demonstrated their two newest NET/PLUS^{tm} products: the NTS10 Network Terminal Server, a four- or eight-port unit that interfaces RS-232C devices onto the Ethernet LAN, and Multi-Vendor Personal Computer Networking Software which provides file transfer and terminal emulation for over a dozen personal computers. This software when used in conjunction with the Network Terminal Server provides communications between PC-to-host, PC-to-PC, and PC-to-device.

Interlan also demonstrated a complete Ethernet data communications package linking together host computers running UNIX to other data processing equipment on the Ethernet network. Demonstrations included remote login, file transfer, and connection to other communication services.

For further information contact:
Gerald W. Wesel
Interlan, Inc.
617-692-3900

Logical Software, Inc.

SoftShell is a full-screen user interface to UNIX with a screen formatter for building forms or menu interface to applications programs. SoftShell provides command templates for all major UNIX commands, reverse scrolling with full-screen editing and re-execution of commands, as well as facilities for displaying the many tree structures of UNIX, such as directory and file systems.

LOGIX is a relational database management system that is completely integrated with the UNIX operating system. LOGIX runs at the shell level, allowing smooth alternation between UNIX and LOGIX commands and allowing LOGIX commands to be packaged in UNIX shell scripts. LOGIX includes a complete interactive command language, a full-screen relation editor, Q programming language/report writer, query compiler, and C interface. A history of all changes to a relation can be maintained through the use of dated relations and several databases can be accessed simultaneously.

For further information contact:
Ilona A. Lappo
Logical Software, Inc.
55 Wheeler Street
Cambridge, MA 02138
617-864-0137

MASSCOMP

MASSCOMP designs, manufactures, sells, and services high-performance, multi-user, M68010/M68000 UNIX systems to end-users and OEM's in engineering, science, and education. The MC-500 and MC-500 WorkStation are based on a triple-bus architecture which anticipates future technological advances in microprocessors, memory, mass storage, communications, and real time interfaces. The CPU utilizes a proprietary bus connecting the M68010/M68000 CPU to system memory. and floating point and array processors. A performance enhanced Intel Multibus^{tm} supports multiple Independent Graphics Processors and a wide selection of system peripherals, including an intelligent Ethernet^{tm} controller board. For data acquisition, the MC-500 supports the Data Acquisition/Control Processor which controls an enhanced STD bus for high speed data acquisition and control at rates up to 1 megasample per second. The Real-Time UNIX operating system is based on UNIX System III, Berkeley virtual memory and Ethernet enhancements, and MASSCOMP's Real-Time and QUICK-CHOICE^{tm} User Interface enhancements.

For further information contact:
Catherine Pfister
MASSCOMP
One Technology Park
Westford, MA 01886
617-692-6200

Megadata Corp.

Megadata's Series 8000 terminals and systems are designed around the 68000 microprocessor. The 8174 Model 1 and Model 4 are designed utilizing Multibus^{tm} compatible modules and include UniPlus+, a UNIX operating system with real time enhancements and a choice of programming languages.

The 8174 Model 3 controller supports up to 32 Model 8178 terminals over a high performance 1 Mbit/second twinax link. Terminals can be located up to 2000 feet from the controller.

The 8178 terminal offers display height adjustment and rotation. Screen tilt and keyboard angle are also adjustable. The programmable keyboard with up to 155 keys is provided for any application requirement.

For further information contact:
Richard Adams
Megadata Corp.
35 Orville Drive
Bohemia, NY 11716
516-589-6800

Micro Focus, Inc.

Micro Focus is the leading producer of Cobol compilers and programmer productivity tools for business application development. Micro Focus products are available for a wide variety of 8-, 16-, and 32-bit microprocessor-based computers, and they support dozens of standard and proprietary operating systems including UNIX Verson 7, 4.1BSD, Xenix, System III and System V. High performance Level II Cobol is written in the native language of the CPU on which it operates and includes a Native Code Generator for fast program execution. Like Compact Level II Cobol, the compiler is GSA certified to the Federal High Level of the ANSI '74 Cobol standard. FORMS-2 is a user-friendly source code-level, interactive debugging and program analysis tool. Together, the products provide an accessible and interactive alternative to developing applications on the mainframe.

For further information contact:
Dan Fineberg
Micro Focus, Inc.
2465 East Bayshore Rd., Suite 400
Palo Alto, CA 94303
415-856-4161

Microsoft Corp.

Microsoft is the developer of Xenix, the most widely used licensed version of UNIX on microcomputers. Xenix is specifically designed for the commercial marketplace, and has numerous enhancements such as usability and reliability, added functionality, increased system security, MS-DOS compatibility, system administration utilities, and improved documentation. Xenix is consistent across major microprocessors, including Intel's 8086 and 286, Motorola's 68000, National Semiconductor's 16032, and Zilog's Z8000. There are more commercial applications available for Xenix than any other form of UNIX. Microsoft demonstrated Xenix on the Intel 310, the Durango Poppy, the Radio Shack Model 16, and the IBM PC-XT.

For further information contact:
John Ulett
Microsoft Corp.
10700 Northup Way
Bellevue, WA 98004
206-828-8080

Momentum Computer Systems International

Momentum manufactures and markets an entire line of 68000, UNIX-based super-microprocessors. The 32/4 is a second generation professional workstation with 1 Mb of memory, 4 serial ports, a graphics subsystem with a second 68000 and ¼ Mb of memory, a 15" monitor (1025×704), two 5 Mb removable cartridge Winchesters and detachable keyboard. The 32 and 32/E are 2 to 16 user systems with 1 Mb of memory, floppy or streaming tape, one or two 5¼" Winchester drives, and parallel port. Software available includes UNIX System III, RM Cobol, Cogen, SVS Pascal, SVS Fortran, SMC Basic, C, 68000 Assembler, Word Processing, Spreadsheet, and MBSI Accounting packages.

For further information contact:
Albert R. Pfeltz
Momentum Computer Systems International
2730 Junction Avenue
San Jose, CA 95134
408-942-0638

National Semiconductor Corp.

UNIX products include the SY516^{tm}, an 8-user development system which utilizes Genix^{tm}. The Genix operating system, an enhancement of Berkeley 4.1 UNIX, is also available in source form as well as binary. Additional products include cross software for the NS16000 family on VAX, in-system emulators, and 8-bit development systems.

For further information contact:
Dave Vonasek
National Semiconductor Corp.
2900 Semiconductor Dr.
Santa Clara, CA 95051

408-721-5000

NCR Corp.

NCR Tower 1632

(1) Distributed Resource System for Twin Towers.

(2) UNIX Version 7 with application tools:
  a. WORDMARC Office Automation
  b. Ingres Relational database
  c. Multiplan
  d. Color Graphics

(3) UNIX System V with TOWERNET linked to other booths at the conference.

(4) A preview of Power-Fail Recovery in the Tower in the event of power outage.

For further information contact:
M. D. Lambert
NCR Corp.
3325 Platt Springs Rd.
West Columbia, SC 29169
803-796-9250

Netcom Products

Fully integrated micro-computer systems featuring:

- MC68000 CPU
- Multibus
- UNIX

A full complement of peripherals and options:

- 20-40 Mb Winchester Drives
- 84-474 Mb SMD
- ½" magnetic tape
- 50 Mb cartridge storage (25 fixed)
- Floating Point
- Rack mount, tower, cabinet enclosures

For further information contact:
Tom McKee
Netcom Products
430 Toyama Drive
Sunnyvale, CA 94089
408-744-0721

Network Research Corp.

FUSION^{tm} Ethernet software is capable of interconnecting IBM-PC, VAX, 68000, PDP-11 and 8086 processors on the same LAN. FUSION is able to cross over operating system boundaries, linking machines running UNIX, MS-DOS and VMS. Fusion provides the user with the ability to transfer files between systems, to execute commands on a remote host, or establish a virtual terminal where, for example, an IBM-PC under MS-DOS or UNIX can act as a terminal to a VAX running UNIX or VMS.

For further information contact:
Kyle Todd
Network Research Corp.
1964 Westwood Boulevard, Suite 200
Los Angeles, CA 90025
213-474-7717

Officesmiths Inc.

The Officesmith^{tm} — A document management system designed to provide organizations with a solution to the maintenance of information. The Officesmith operates in a structured environment which includes multiple windows, menus, single function key commands and a user programmable help facility, and supports a complete range of document management activities. Combining powerful yet simple application development tools and document management facilities, The Officesmith offers systems integrators and office systems developers the resources to build and maintain responsive corporate information structures.

OfficePolicy — A methodology for documenting, communicating and accessing policies and procedures in large organizations. OfficePolicy is a complete approach combining management guides, training courses, and software tools based on The Officesmith^{tm}.

For further information contact:
Glenn A. McInnes
Officesmiths Inc.
331 Cooper Street
Ottawa, Ontario K2P 0G5 Canada
613-235-6749

Onyx Systems, Inc.

Onyx Systems exhibited UNIX-based microcomputer systems first introduced at Comdex Fall. The model C5012D is a desk-top system with 512 Kbytes of RAM, 14 or 21 Mbyte Winchester disk and tape cartridge backup. It supports five users under UNIX System III. The C5012V is a floor-standing unit with up to one megabyte of memory and up to 42 megabytes of Winchester storage. The systems are bundled with the Onyx Office, an integrated applications package including word processing, spreadsheet and database manager

For further information contact:
Tom Anthony
Onyx Systems, Inc.
25 E. Trimble Road
San Jose, CA 95131

Oregon Software, Inc.

Oregon Software exhibited their Pascal-2 compiler system. This includes a high-performance compiler for the MC68000 and PDP-11 running on UNIX. A source-level, interactive debugger, performance profiler, and utility package are also included.

The MC68000 UNIX compiler executes on the UniSoft, UniPlus+ system and Xenix. End use pricing for Pascal-2 on the MC68000 is $1,650 including one year of support. OEM prices available upon request.

For further information contact:
David Cloutier
Oregon Software, Inc.
2340 SW Canyon Road
Portland, OR 97201
503-226-7760

Pacific Microcomputers, Inc.

Pacific Microcomputers' display consisted of several levels of 68000/68010 single board microprocessors displaying the latest technology in controlling multiprocessor environments, high-speed data transfer, and no-wait-state processing for OEM applications. In addition, Pacific exhibited full integrated microprocessors with UNIX operating system and full support of all standard programming languages. These systems are highly flexible designs utilizing both 5¼″ and 8″ technology providing high performance aimed at the multi-user marketplace for OEM's and sophisticated end users.

For further information contact:
Sherrell Harper
Pacific Microcomputers, Inc.
119 Aberdeen Drive
Cardiff, CA 92007
619-436-8649

Paradyne Corp.

Paradyne's exhibit included demonstrations and presentations of a UNIX-derived communications Terminal Controller. Supporting multiple protocols and multiple terminal personalities two models were displayed. A "desk top" System 8400 supports up to 8 terminals and/or printers with 4 communications ports and a 26 Mb disk and a 640 Kb diskette. The application development System 8400 supports up to 16 terminals and or printers with a 9 track 1600 bpi tape and 160 Mb disk. Both systems use a UNIX-derived operating system and a combination of microprocessors the Z8000 and the 80186 for the CPU and the communications controller respectively. Paradyne is a communications corporation and has over 1900 System 8400's installed.

For further information contact:
Elliott Kane
Paradyne Corp.
PO Box 2826, 8550 Ulmerton Road
Largo, FL 33540
813-530-2577

PHACT Associates Ltd.

PHACT Associates provide a range of software tools aimed at the applications builder and end-user.

Products include: PHACT-dbrm, a multi-keyed ISAM-like database record manager, incorporating a data dictionary, variable length records etc; PHACT-rql, a relational query and data manipulation language; PHACT-rg, a report generator.

New products include SaPHACT, a UNIX filter for manipulating PHACT databases. Products under development include a list manager and an object manager.

For further information contact:
David Graham
PHACT Associates Ltd
225 Lafayette Street
New York City, NY 10012
212-219-3744

Plexus Computers, Inc.

Plexus initiated its "take the Plexus challenge" program and demonstrated a local area network including the Plexus P/65 network file server, two IBM PC's as UNIX-based workstations, and a Plexus P/35 32-bit supermicrocomputer.

Under the "take the Plexus challenge" program, attendees ran benchmark studies on a Plexus P/35.

The most recent Plexus computer developed, the P/65, offers up to 1.1 Gb of storage and serves as a database management and network management tool. In combination with the Unify DBMS, it provides fully transparent multi-user access to shared data on an Ethernet-based network. IBM PC's were linked to the network as standalones or UNIX-based workstation through Lantech systems' Unetix software.

For further information contact:
Brenda Birrell
Plexus Computers, Inc.
2230 Martin Avenue
Santa Clara, CA 95050
408-988-1755

Productivity Products International

Productivity Products has a product name Objective-C. This product brings the advantages of object oriented programming to the C language. We are producing Tool Kits for professional programmers and coordination tools for systems houses.

For further information contact:
Andrew Iorio
Productivity Products International
Newton, CT 06482
203-426-1875

Proper Software

We sell DRIVER, a visual replacement for the *cd* command, to OEM's.

For further information contact:
Paul Hoffman
Proper Software
2000 Center Street, Suite 1024
Berkeley, CA 94704

Quadratron Systems, Inc.

Q-Office is an office automation software package which includes Q-One, word processing; Q-Date, electronic calendar; Q-Menu, menu generator; Q-note, electronic note pad; Q-Form, form generator; Q-Call, phone directory; Q-Math, calculator; and Q-Mail, electronic mail. Q-Office is completely device independent and comes with *termcap* and *printcap* files and generators.

For further information contact:
Vanessa Abbe
Quadratron Systems, Inc.
15760 Ventura Boulevard, #1032
Encino, CA 91436
818-789-8588

Relational Technology, Inc.

Relational Technology Inc. displayed INGRES, a relational database management and applications development system. INGRES provides flexible report writing capabilities; data management and applications generation through Visual Programming^{tm} forms interfaces, and a powerful query language, QUEL, for interactive use or easy embedding in most programming languages. INGRES includes INGRES/REPORTS^{tm}; INGRES/QUERY^{tm}; INGRES/FORMS^{tm}; INGRES/APPLICATIONS-BY-FORMS^{tm}; INGRES/GRAPHICS^{tm}; INGRES/NET^{tm} for distributed access; and EQUEL (Embedded QUEL) interfaces for Fortran, Cobol, Basic, Pascal, and C languages. INGRES runs on VAX computers under the UNIX or VMS operating system, or on MC68000-based supermicrocomputers under UNIX operating systems.

For further information contact:
Peter Tierney
Relational Technology, Inc.
2855 Telegraph Avenue
Berkeley, CA 94705
415-845-1700

Sage Computer

The Sage II and the Sage IV supermicros featured the 16/32-bit MC68000 microprocessor. Memory ranges from 256 Kb to 1 Mb with storage options from one 640 Kb floppy to Winchester hard disks up to 40 Mb. All systems can be configured for multi-user operation with each user having a partition of memory within the main processor. All Sage Computer systems run the Idris operating system from Whitesmith, Ltd. and Logos Information Systems.

For further information contact:
Bill Delaney
Sage Computer
4905 Energy Way

Reno, NV 89502
702-322-6868

The Santa Cruz Operation

The Santa Cruz Operation (SCO) features the Xenix operating system and a full range of application software for UNIX-based systems. SCO's selection includes the Multiplan^{tm} spreadsheet package; Uniplex^{tm}, a comprehensive, menu-driven word processor; fully GSA certified Level II Cobol^{tm}; and the Informix^{tm} relational database management system. The company also features the UNIX system Tutorials, an introductory training package for new UNIX system users; an integrated family of cross-assemblers; and other UNIX-based products. For the UNIX-system OEM, SCO provides comprehensive system support and software development services.

For further information contact:
Doreen Hamamura
The Santa Cruz Operation
500 Chestnut Street
Santa Cruz, CA 95060
408-425-7222

Silicon Graphics, Inc.

Silicon Graphics demonstrated the IRIS (Integrated Raster Imaging System) workstation, with its real-time 2D and 3D graphics capability. The IRIS uses the Geometry Engine^{tm}, a proprietary VLSI chip designed by Silicon Graphics, to provide real-time graphics processing. This chip is integrated into a system of innovative hardware, powerful graphics software, the UNIX operating system, and the Ethernet communications network. The IRIS provides a powerful general-purpose computing environment suitable for complex engineering and scientific applications.

For further information contact:
Randy Nickel
Silicon Graphics, Inc.
630 Clyde Court
Mountain View, CA 94043
415-960-1980

SKY Computers Inc.

SKY is in the business of providing state-of-the-art high-speed arithmetic peripherals. The current product line includes a Fast Floating Point Processor (SKYFFP), and a 1 MFlop array processor (SKYMNK). The SKYFFP provides high-speed floating point arithmetic to M68000 based systems on four different standard buses as well as some proprietary buses. It is totally transparent to the user and provide many compound functions in microcode for higher speed. The SKYMNK is a board-level array processor for integration into most popular microcomputers including Q-bus, Multibus (Intel and M68000) and Versabus based systems as well as some proprietary buses. The SKYMNK comes with a very rich set of subroutines for optimal array processor use.

For further information contact:
Howard Klemmer
SKY Computers Inc.
Foot of John Street
Lowell, MA 01852
617-454-6200

Spectrix Inc.

Spectrix Inc. is a designer, manufacturer and marketer of hardware, operating systems and software. Our S10 and S30 are high performance product families based on MC68000 technology, run under UNIX and use multi-buses. In addition we have created Basic 3 which is a Dartmouth Basic language that features virtually 100% compatibility with Wang Basic 2. Basic 3 gives Spectrix users access to a library of thousands of application software programs that have been proven over the years on Wang.

For further information contact:
Richard C. Drew
Spectrix Inc.
3000 Dundee Road
Worthbrook, IL 60062
312-291-0850

Sritek Inc.

Sritek manufactures high-performance coprocessors and advanced operating systems for the IBM PC and XT. The Xenix System 3.0 for the Sritek 6800 coprocessor offers paged-memory management which can support up to 16 users. For the 16032 coprocessor we offer the Berkeley 4.1BSD which supports 32-bit demand-paged virtual memory management and floating-point hardware. We will soon offer the 4.2BSD version with networking facilities. UNIX System V will be available on the 68000, 16032 and 80286 coprocessors during the second quarter of 1984.

For further information contact:
Walter Fuchs
Sritek Inc.
10230 Brecksville Road
Cleveland, Ohio 44141
216-526-9433

SofTest Inc.

SofTest exhibited its four UNIX-based software packages: SofGram — the only UNIX product to transmit and receive messages over the Telex, TWX and phone networks; LEX word processing system; The Menu System — a user friendly front-end to any UNIX based application that allows a designer to build window-based menus in English; and SofForms a forms creation and management package.

For further information contact:
Michael Heffler
SofTest Inc.
555 Coggle Road
Ridgewood, NJ 07450
201-447-3901

Software Express

"APPGEN" – Applications Generator: First non-procedural 4th generation UNIX applications development and maintenance environment that produces applications without coding.

Produces: Transaction driver, interactive, commercial application with a limitless number of files, screens, menu's and reports.

Complete Environment Includes: Screen creator, report creator, Relational DBMS, Parametric Definition Methodology, Automatic end-user documentation, English Query, interactive structured design facility, optimized C run time module, import export facility.

Benefits: Develop 10 times faster, modify 50 times faster. Operating Systems: UNIX (3.0, 4.2, 5.0, 7.0), Xenix, Pick. Cost: $6,000 Development, $600 run time. Applications Available: G/L, A/P, A/R, Payroll, inventory, plus 4 verticals. Support: training, maintenance, video tapes, professional documentation.

For further information contact:
Steve Thomas
Software Express
10103 Fondren, Suite 220
Houston, Texas 77096
713-220-5218, 800-231-0062

Sphinx Ltd.

Sphinx is a software marketing company specializing in products related to UNIX, the standard multi-user operating system for the latest micro and mini computers. Sphinx was set up in May 1983 to provide marketing and service links between developers of multi-user software and potential professional users. Sphinx acts as an independent supplier qualified to advise users on suitable products for their applications needs. Sphinx offers software developers and suppliers professional marketing skills and services. These include specification and testing of software, documentation and packaging, promotional activities and sales. Sphinx guarantees a high level of service throughout it's sphere of operations.

For further information contact:
Cornelia Boldyreff
Sphinx Ltd.
43-53 Moorbridge Road
Maidenhead, Berkshire
SL6 8PL England
+44 628 75343

STSC, Inc.

STSC has developed a high-performance enhanced APL interpreter for the UNIX environment to complement it's very popular APL*PLUS/PC system and it's APL products for IBM mainframes. The APL*PLUS/UNIX system runs on a wide variety of supermicro and mini computers that use UNIX System 3 or System V. This all-new APL system provides features that previously have been available only on mainframe systems – including rested arrays and very large APL variables and workspaces.

For further information contact:
Rich Paulson
STSC, Inc.
21125 E. Jefferson Street
Rockville, MD 20852
301-984-5000

Texas Instruments Inc.

Nu Machine Computer Family

This computer was architected at the Laboratory for Computer Science at MIT and is based on the MIT Nubus. Two versions were announced Jan. 10 1984 and shown for the first time at UniForum '84. This high-end UNIX oriented product supports one or two high resolution bit mapped display and is available with an 84 or 474 Mbyte disk. A two user configuration provides high resolution displays at under $20,000 per user in OEM quantities.

For further information contact:
George White
Texas Instruments Inc.
17781 Cartwright Road
Irvine, CA 92714
714-660-8243

Tom Software

Tom Software is a software manufacturer distributing its products through a world wide network of licensed consultants. Tom provides applications and utility solutions for the first time user and data processing professional. The applications include distributors, residential contractors, manufacturers, specialty contractors, general contractors, property management, restaurants, not for profit organizations word processing, and an applications development utility.

For further information contact:
Jerry Hill
Tom Software
PO Box 66596
Seattle, WA 98166
206-246-7022

TYX Corp.

TYX Corporation, a vendor of typesetting software and typesetting hardware packages, presented two of their systems, the TYXSet 1000 Textsetting System and the TYXSet 100 Personal Textsetting System. The software runs on UNIX System III, V, and Xenix. It includes the TEX typesetting language written in C, a user-friendly menu package for typesetting composition using TEX, and a powerful text editor for creating and editing documents.

The TYXSet 1000 package is a multi-user system designed especially for agencies with high volume print output. It includes the TYXword/T software system, the T/10 laser printer, and the T/25 computer. The TYXSet 100 package is a single user system which includes a Victor 9000 personal computer equipped with a 10 Mb hard disk, a text editor for document fabrication, the TEX typesetting menu driven package, and a high resolution bit-mapped proofer.

For further information contact:
James Gauthier
TYX Corp.
11250 Roger Bacon Drive, Suite 16
Reston, VA 22090
703-471-0233

UniComp Corp.

UniGen is an application generator for the UNIX operating system, UniGen allows the generation of UNIX applications using menus, extensive user prompting, and maintenance of random access record files.

For further information contact:
Donald Anderson
UniComp Corp.
202 Plaza Towers
Springfield, MO 65804
417-883-6800

Unify Corp.

Unify is a multi-user, menu driven and fully interactive UNIX relational data base management system. It is designed to support large data bases up to 2 billion records over 8 physical devices.

Unify provides a complete range of application development tools including a menu handler, automatic data entry, query by forms, report generator, and structured query language (SQL) based on IBM's SEQUEL 2 relational compatible with C and RM-Cobol, is also provided.

Unify offers multiple data access and raw file I/O capability for fast performance while only requiring a minimum memory configuration of 256K bytes and 2 megabytes of disk space.

For further information contact:
Duke Castle
Unify Corp.
9570 SW Barbur Boulevard, Suite 303
Portland, OR 97219

UniPress Software, Inc.

UniPress Software, Inc. exhibited its UniPlus+ UNIX System V operating system for the Apple Lisa Computer, as well as other UNIX software available for the VAX and many 68000 implementations. Products on display were EMACS, a full screen multi-window editor with built-in MLISP programming language, LEX word processing system, a menu driven interactive word processor with full screen editing and scrolling, the highly regarded Lattice C compiler, both native and cross for UNIX and MS-DOS environments. Other products on display included PHACT ISAM, /ROB, UniCalc spreadsheet, mimix CP/M emulator, and the menu system.

UniPress Software is a software publishing firm for UNIX, VMS and MS-DOS environments. Our UNIX software is available on MC68000 implementations including the Sun, MASSCOMP, Plexus, Dual, Tandy 16, Apple Lisa, Apollo, etc. as well as the VAX computer.

For further information contact:
Joyce Bielen
UniPress Software, Inc.
1164 Raritan Avenue
Highland Park, NJ 08904
201-985-8000

Uniq Computer Corp.

Uniq Computer Corporation announced UNIQORN, a packaged system based on the long-awaited Digital LSI-11/73, the single-board implementation of the PDP-11/70 CPU. UNIQORN offers PDP-11/44 class performance at a much lower price. In addition, Uniq provides UNIX System V, System III and Berkeley implementations on Digital VAX and PDP-11 computers. A variety of software products are available, including UNIFY DBMS, UNICALQ virtual memory spreadsheet, RM/Cobol, Basic, Pascal, Sibol, word processing and networking. Uniq provides ongoing support for all of its products as well as consulting and training the UNIX environment. Training is held in both scheduled classes and onsite at client locations.

For further information contact:
Roger A. Knuth
Uniq Computer Corp.
28 S. Water Street
Batavia, IL 60510
312-879-1008

Unisource Software Corp.

Unisource Software Corp. is a major publisher and distributor of UNIX software for the corporate, professional and system development markets. The company's UniForum exhibit featured its Office UNIX system, a package consisting of a variety of business applications running under VENIX/86, a licensed implementation of AT&T's UNIX operating system. Single and multi-user configurations of IBM PC's, Compaqs and display terminals were shown. Fusion Ethernet software newly ported to the IBM PC was shown for the first time, linking Unisource's system with those of several other exhibitors. Among the UNIX-based applications available form Unisource are The FinalWord word processor; Leverage for database management; ViewComp, an electronic spreadsheet; Sunburst accounting packages; and the company's own Office Menu Tool designed to make UNIX easy to use for inexperienced users.

For further information contact:
Dena Wortzel
Unisource Software Corp.
71 Bent Street
Cambridge, MA 02141

UniWare (Division of Nuvatec, Inc.)

UniWare offers UNIX-based microprocessor software cross-development tools. Approximately 10 popular UNIX-based hosts and 20 target microprocessors are currently supported; support for others is announced regularly. A cross-development package consists of a macro preprocessor, cross-assembler, link editor, downloaders, "generic" downloader source code, and various utility programs. C cross-compilers are available for selected target processors.

Significant new product announcements will be made later this year.

For further information contact:
Eric Thiele
UniWare (Division of Nuvatec, Inc.)
261 S. Eisenhower
Lombard, IL 60148
312-620-4830

Vandata

Vandata is a software firm that specializes in computer languages, both as commercial products and as a contract service to industry. Since 1975, Vandata has been responsible for the development of a number of interpreters and compilers as well as two parser generators, a C subset for small ROM-based systems and a Whitesmiths-based Z80 C cross-compiler.

The Vandata Z80 C compiler is a full implementation of the C language, targeted for the Z80 microprocessor. The compiler can be run on PDP-11, VAX, Z8000, and 68000 hosts, as well as on CP/M systems. Based on Whitesmiths' compiler, Vandata C offers thorough support for ROM/RAM environments, extremely compact generated code, and the ability to intermix assembly language routines with C routines. An optimizer, Z80 assembler, linker, librarian, and other tools are provided with the compiler, as well as the facility to interface to other cross-assemblers. A royalty-free runtime library is available.

For further information contact:
Dwight Vandenberghe
Vandata
17544 Midvale Avenue North, Suite 107
Seattle, WA 98133
206-542-7611

Voelker Lehman Systems, Inc.

Voelker Lehman is a hardware/software systems integrator with a variety of products and services including:

- VAX-based multi-vendor hardware systems:
- Hardware configuration, integration, installation and support;
- Turbo-VX, performance accelerator for VAX 11/750;
- U/OS, enhanced UNIX operating system software compatible with Berkeley 4.1 featuring System V enhancements and U/INSTAL, automatic installation utility;
- U/CDB, Interactive Source-Level C language debugger;
- Vanilla 4.1 and 4.2BSD;
- Full Software Support

For further information contact:
Joe L. Voelker
Voelker Lehman Systems, Inc.
4400 Old Warm Springs Boulevard
Fremont, CA 94538
415-490-3555

The Wollongong Group

Eunice furnishes VAX users with the ability to merge the VMS environment with the advantages of UNIX commands and utilities. Eunice permits users to transport previously written software from either the VMS or UNIX domain, without affecting the software. Eunice allows any mixture of VMS and UNIX users to share the resources of a single computer system.

PEGASUS off-loads context switching tasks associated with terminal I/O communications in a keyboard transparent environment. PEGASUS off-loads either *vi* or Wollongong's Full Screen Editor for host systems running VMS, UNIX, or Eunice.

Wollongong's other system level software products include: RM/Cobol, Mistress, Horizon, Full Screen Editor, and IP/TCP. Wollongong markets consulting and engineering services to other UNIX companies, and major hardware manufacturers desiring to participate in the UNIX marketplace. These services include performance analysis, custom hardware/software integration, and marketing assistance with new UNIX products. Educational courses and seminars are available for new and experienced UNIX users.

For further information contact:
David J. Preston
The Wollongong Group
1129 San Antonio Road
Palo Alto, CA 94303
415-962-9224

Yates Ventures

Yates Ventures conducts primary market research and performs hands-on product evaluations of UNIX hardware and software. Multiclient subscribers receive definitive market data and analysis as an aid in strategic decision making. The Yates Perspective, a monthly newsletter, keeps clients up to date on events in the UNIX market. The UNIX Encyclopedia is a popular end-user source of UNIX product information.

For further information contact:
Pamela Pasotti
Yates Ventures
4962 El Camino Real, Suite 111
Los Altos, CA 94022
415-984-0130

USENIX Association
Application for Membership
Calendar Year 1984

Please type or print

[ ] New [ ] Renewal - Member number from mailing label: __________

Member Name: ______________________________

Address*: ______________________________

____

____

____

Phone: ______________________________

uucp network address: ______________________________

Name and address of official Member Representative, if membership is for an institution:

____

____

____

Phone: ______________________________

uucp network address: ______________________________

Membership Fees:

[ ] $ 30 Individual

[ ] $ 250 Institutional

[ ] $1000 Supporting

[ ] $ 9 Overseas airmail
(Fee required for Individual or Student Memberships only)

[ ] $ 15 Student (full-time)

[ ] $100 Institutional - educational discount
open to accredited educational institutions

Individual, Institutional, and Supporting categories are all open to either institutions or individuals.

[ ] Check enclosed: $__________ [ ] Purchase order enclosed; invoice required
Payments must be in US dollars payable on a US bank

[ ] Check here if you do **NOT** want your name and address made available to other members, either verbally or in printed matter.

[ ] Check here if you do **NOT** want your name and address made available for commercial mailings.

Inst: ....
Del: ....
Mem#: .... Check#: ....
Lic: ....
Date: ....
Db: ....

---

* Accounting and billing address, if membership is for an institution

OVER

For Institutional and Supporting Memberships Only

Institutional and Supporting Members with UNIX licenses may take advantage of certain services requiring license verification. Please indicate the type of UNIX license(s) held by writing **S** (for source) or **B** (for binary):

___Mini-UNIX	___V6	___PWB	___Phototypesetter	
___V7	___32V	___System III	___System V	___UNIX/UNIVAC
___UNIX/TSS	___UCB 4.xBSD	___UCB 2.xBSD	___Other__________	__________

If you do hold UNIX licenses, please send the signature page and the pages that show the version of UNIX you are licensed for, whether it is source or binary, the name of the institution owning the license, and the type, serial number, and location of the CPU(s). If you have more than one license, please send the above information for each.

Institutional or Supporting members renewing their memberships need send only newly acquired licenses or those not previously verified by USENIX.

Institutional and Supporting Members (whether new or renewing) must also complete a USENIX **1984 Tape Release Form** if they wish to receive distribution tapes.

[ ] Licenses(s) enclosed [ ] Licenses(s) already on file

[ ] Tape Release enclosed

Authorized Signature: ______________________________ Date: ____________

Return completed form to:

USENIX Association
P.O. Box 7
El Cerrito, CA 94530

This form **must** accompany your purchase order or payment. You will receive a card acknowledging your membership as soon as it is processed.

Release Form for 1984 USENIX Distribution Tapes

Institutional or Supporting Members Only

As an Institutional or Supporting Member of the USENIX Association, I wish to receive the 1984 USENIX Distribution Tape(s) (hereafter referred to as the TAPES). By virtue of this signed release form, I agree to the following Terms and Conditions governing the use of any and all TAPES sent to me.

(1) **Neither the USENIX Association, its officers, employees, and agents, nor the authors nor submitters of the materials on the TAPES (hereafter referred to as the ABOVENAMED) make any representation or warranty, expressed or implied, as to the accuracy, completeness, or usefulness, of the TAPES or the contents thereof for any purpose whatsoever.**

(2) To the extent permitted by law, I hereby explicitly indemnify and hold harmless the ABOVENAMED from any claims, costs, damages and liability whatsoever with respect to the TAPES or the contents thereof.

(3) No property rights with respect to the TAPES or the contents thereof shall transfer to me. The TAPES shall remain the property of the USENIX Association and the contents thereof shall remain the property of the person or persons who possessed that property before the TAPES were written.

(4) The licensing of my use of the materials on the TAPES by the USENIX Association is limited to those materials for which the USENIX Association is authorized to do so. The USENIX Association has attempted to verify that it has the right to distribute all of the materials on the TAPE, but it cannot guarantee that it has such right. In the event that any dispute arises as to the distributed materials, I agree that the USENIX Association may direct me to erase materials from the TAPES and that I will remove said materials from the TAPES, destroy all copies thereof, and make no claim against the ABOVENAMED for any action arising out of the use or the destruction of the materials.

(5) I certify that I have the UNIX licenses or sub-licenses indicated below and that I shall use the materials on the TAPES so as not to violate said licenses. In particular, I agree that if there should be material on the TAPES that I am not licensed under our UNIX license(s) to receive or use, I shall neither use nor make copies of said materials.

(6) I agree that this license grants me no rights to subdistribute any or all of the materials on the TAPES, except for use at the address designated below or on my membership form. This license grants me no rights to incorporate any of the materials on the TAPES in a commercial product, and I agree that, if such materials are not in the public domain, I will secure such rights from the owner of the property before incorporating any of the materials on the TAPES in a commercial product.

Subject to the above conditions, the USENIX Association grants and I hereby accept a **limited license to use** the materials on the TAPES at the address given below.

The contributions are generally separated into distribution tapes containing material covered by different licenses. Tapes will include all material from earlier versions of licenses that the current license specifically allows (e.g. the System III tape will include material licensed for V6, V7, etc. as well as no-disclosure material). **Binary license holders may receive ONLY the tape with no disclosure material.**

As an Institutional or Supporting Member, you are entitled to one tape per distribution. (However, additional different versions, for which you have appropriate licenses, may be purchased. Contact the USENIX Office for more information.) Since there is no guarantee that material will be submitted for all license categories, please indicate the UNIX licenses you own and your preference for receiving tapes (1 = first choice, 2 = second, etc.):

	Source License	Binary License	Tape Preference
System V	☐	☐	____
System III	☐	☐	____
UNIX Version/32V	☐	☐	____
UNIX Version 7	☐	☐	____
PWB/UNIX Version 1.0	☐	☐	____
UNIX Version 6	☐	☐	____
Other: ____________	☐	☐	____
No-Disclosure tape			____

Tape Density (specify only if you cannot accept either density):

☐ 800 bpi ☐ 1600 bpi

Address to which your TAPES should be sent and where TAPES will be used, if different from address on membership form:

Institution/Member Name: ________________________

please print

Member number (if known): __________

Authorized Signature: ____________________ Date: ________

USENIX Association Signature: ____________________ Date: ________

Copies of your relevant UNIX license(s) must be on file at the USENIX Office for you to receive your tape. It is your responsibility to notify the USENIX Office should there be any change in your licensing categories, tape preference, or mailing address. Please sign and return **two** copies of this form to the USENIX Association Office. One copy will be signed by a USENIX representative and returned to you with your TAPES.

For Office Use Only

lic __________ how ________ date ________ by ______

type sent ______ date ________ by ________ db ______

This page is blank

USENIX Association
P.O. Box 7
El Cerrito, CA 94530

First Class Mail

EL CERRITO CALIF.
MAR 17'84
U.S. POSTAGE
1.05
P.B. METER
7226856

Reports on Presentations at UniForum

Summer '84 Meeting Call for Papers